COMPUTING FOR HOME AND BUSINESS

INTERFACE AGE™

02651 APRIL 1980 $2.00

DEFINING ARTIFICIAL INTELLIGENCE

Promotional Androids

Robot Guidance

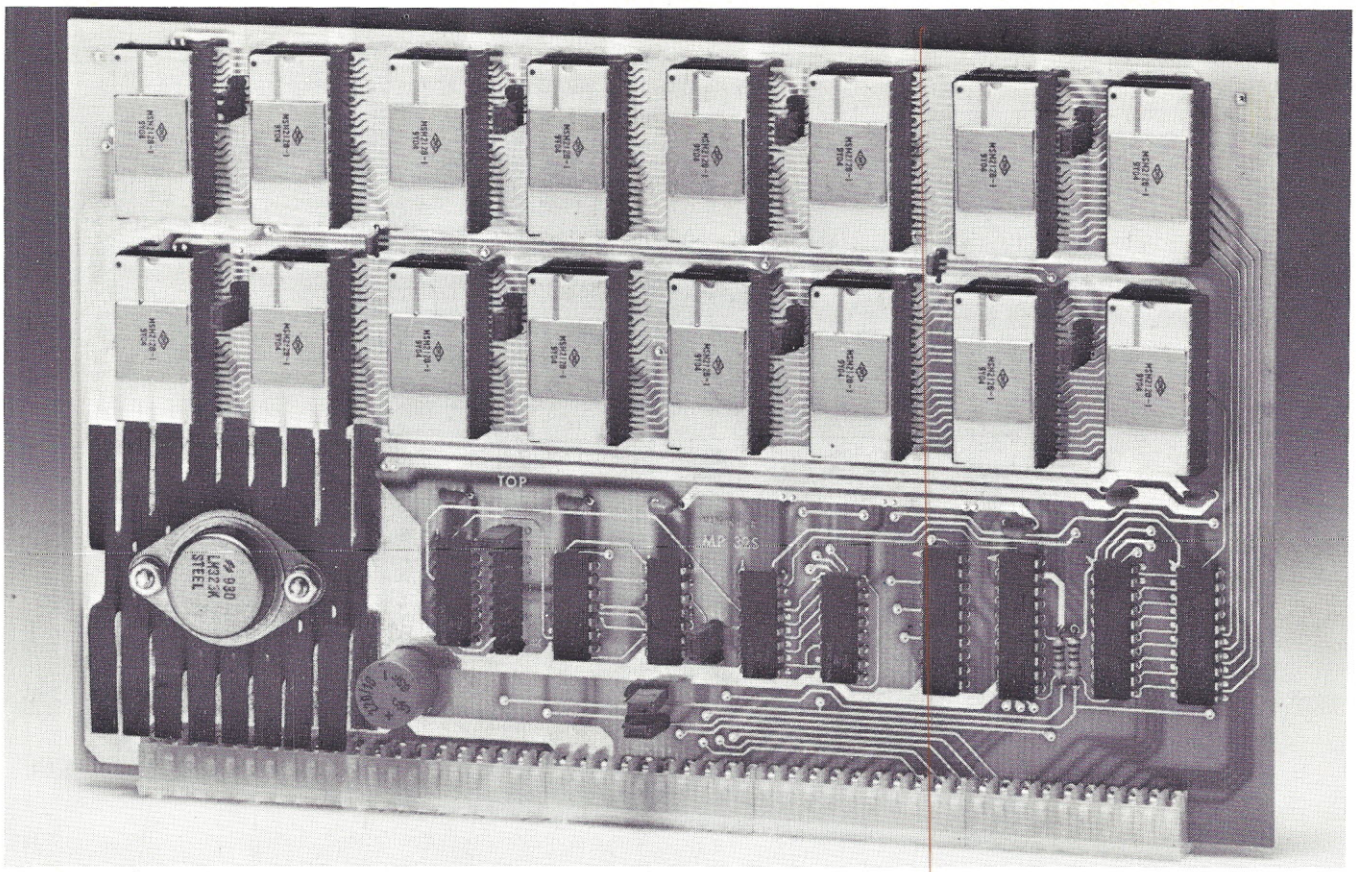
Using & Building Micro-Based Systems - Part IV



...M THE LIBRARY
Wichita
TX 76308
WICHITA FALLS
3400 TAFT ST
MIDWESTERN ST UNIV LIB
X 0564332 8106 SSSS
1

0 71486 02651 04 DIR

CANADA/MEXICO $2.50



UNIVERSAL static memory card

★ 32K bytes-ROM, RAM, EPROM or a combination
★ SS-50 A&C compatible with 16 and 20 bit address decoding
★ Compatible with all SWTPC 6800 and 6809 computers

This is the most versatile memory card you can buy. Our S-32 may be populated with up to 32K of static RAM, EPROM, or ROM, or any 4K block combination of these that you may desire. Any 5-volt 2716 pinout compatible memory may be used in this card. Any 4K block of the memory may be jumper block programmed for RAM or ROM use. This feature makes this the ideal memory for those process control applications that require a mixture of ROM and RAM memory. The board is fully compatible with all SWTPC 6800 and 6809 computers.

The power requirement for the board is only 1.75 amps at 5.0 volts with a full 32K of RAM installed.

S-32 Circuit card assembled
—less memory IC's (uses up to 16).$99.50

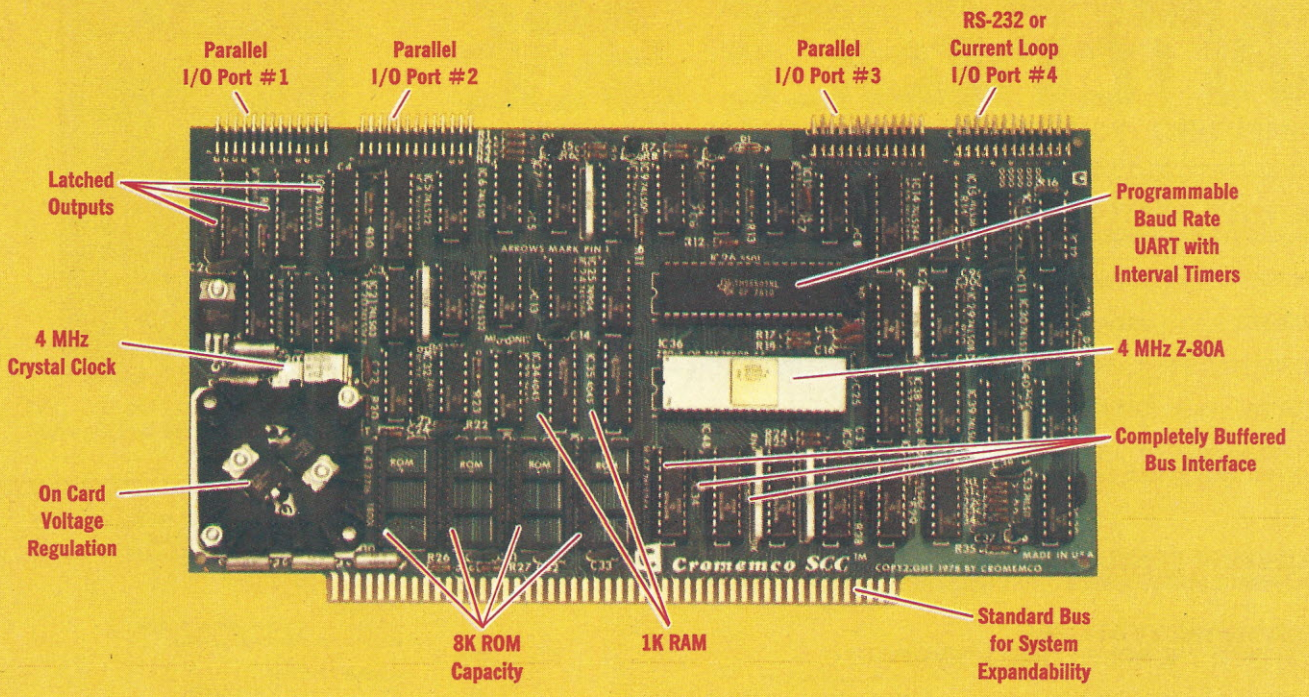
2716 Type EPROM for above $50.00 ea.

16K (2K x 8) Static RAM for above
(4016 or 2128). $50.00 ea.

SWTPC

SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY
SAN ANTONIO, TEXAS 78216 (512) 344-0241

CIRCLE INQUIRY NO. 64



The single card computer with the features that help you in real life

COMPLETE COMPUTER

In this advanced card you get a professional quality computer that meets today's engineering needs. And it's one that's complete. It lets you be up and running fast. All you need is a power supply and your ROM software.

The computer itself is super. Fast 4 MHz operation. Capacity for 8K bytes of ROM (uses 2716 PROMs which **can be programmed by our new 32K** BYTESAVER® PROM card). There's also 1K of on-board static RAM. Further, you get straightforward interfacing through an RS-232 serial interface with ultra-fast speed of up to 76,800 baud — software programmable.

Other features include 24 bits of bi-directional parallel I/O and five on-board programmable timers.

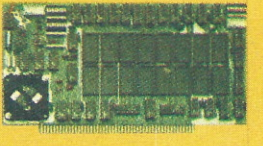
Add to that vectored interrupts.

ENORMOUS EXPANDABILITY

Besides all these features the Cromemco single card computer gives you enormous expandability if you ever need it. And it's easy to expand. First, you can expand with the new Cromemco 32K BYTESAVER PROM card mentioned above. Then there's Cromemco's broad line of S100-bus-compatible memory and I/O interface cards. Cards with features such as relay interface, analog interface, graphics interface, opto-isolator input, and A/D and D/A conversion. RAM and ROM cards, too.



Card Cage



32K BYTESAVER PROM card

EASY TO USE

Another convenience that makes the Model SCC computer easy to use is our Z-80 monitor and 3K Control BASIC (in two ROMs). With this optional software you're ready to go. The monitor gives you 12 commands. The BASIC, with 36 commands/functions, will directly access I/O ports and memory locations — and call machine language subroutines.

Finally, to simplify things to the ultimate, we even have convenient card cages. Rugged card cages. They hold cards firmly. No jiggling out of sockets.

AVAILABLE NOW/LOW PRICE

The Model SCC is available now at a low price of only $450 burned-in and tested (32K BYTESAVER only $295).

So act today. Get this high-capability computer working for you right away.

Cromemco
incorporated
Specialists in computers and peripherals
280 BERNARDO AVE., MOUNTAIN VIEW, CA 94040 • (415) 964-7400

VOL. 5, ISSUE 4
SINCE DECEMBER 1975

APRIL 1980

INTERFACE AGE™

COMPUTING FOR HOME AND BUSINESS APPLICATIONS

GENERAL FEATURES

PROMOTIONAL ROBOTS: More Than Mechanical Toys 66
by Kathy Tekawa, Assistant Editor

AN INEXPENSIVE ROBOT GUIDANCE SYSTEM: 'EYES' FOR YOUR ANDROID 70
by Dave L. Morris

ROBOTICS AND URBAN TRAFFIC 72
by Boris Dobrotin

COMPUTER SIMULATION OF CLASSICAL AND INSTRUMENTAL CONDITIONING . 74
by Alan Filsinger

CAN YOUR ROBOT PASS THE TEST? 83
by Alan Thompson, Robotics Age

A DISCUSSION OF ARTIFICIAL INTELLIGENCE, A DEFINITIVE ANSWER TO THE QUESTION: CAN COMPUTERS THINK? 85
by Roger C. Garrett, Contributing Editor

BUSINESS FEATURES

DATABASE MANAGEMENT FOR THE APPLE II 94
by Phil Roybal, Apple Computer, Inc.

HARDWARE FEATURES

SYSTEM OF THE MONTH: OHIO SCIENTIFIC C3-C 102
by Tom Fox, Systems Editor

POWERED DOWN BIPOLAR PROMS, A 'COOL' OPERATING SYSTEM 108
by John MacDougall

LEARNING CENTER

MY TRS-80 LIKES ME 113
by Bob Albrecht

TUTORIALS

THE PASCAL NOTEBOOK, CHAPTER 9 115
by Henry Davis

USING AND BUILDING MICRO-BASED SYSTEMS 122
by David Marca, Associate Editor

SOFTWARE FEATURES

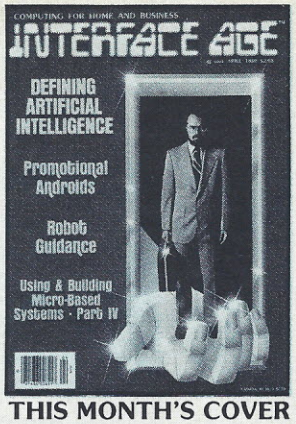
AN 8080/Z-80 DISK BASIC, MICROSOFT'S VERSION 5 118
Review by Alan R. Miller, Software Editor

FREE RUNNING

AL BAKER'S GAME CORNER 37
BUSINESS SOFTWARE REVIEW 62
THE COLUMN 24
EDITOR'S NOTEBOOK 4
FROM THE FOUNTAINHEAD 30
INDUSTRY MEMO 58
INVENTOR'S SKETCHPAD 59
JURISPRUDENT COMPUTERIST 34
LETTERS TO THE EDITOR 8
MICRO MATHEMATICIAN 42
MIND REVOLUTION 56

DEPARTMENTS

ADVERTISER INDEX 144
BOOK REVIEWS 126
CALENDAR 21
MICRODEX 143
MICRO MARKET 142
NEW PRODUCTS 128
UPDATE 19



THIS MONTH'S COVER

Design and art by Fino Ortiz, Art Director; photography by Don May. Model, Mike Antich.

16704 Marquardt Avenue, Cerritos, CA 90701
(213) 926-9544
TWX (910) 583-1412

PUBLISHERS
PUBLISHER/EDITOR-IN-CHIEF **ROBERT S. JONES**
EXECUTIVE PUBLISHER **NANCY A. JONES**

ADMINISTRATION
PUBLICATION DIRECTOR **MIKE ANTICH**
PUBLICATION ASSISTANTS **DORIS RIOPEL, CHERYL JOHNSTON**
ACCOUNTING SUPERVISOR **KAY SOTO**
ACCOUNTING ASSISTANTS **MARY ANN LOWER, SHIRLEY MAZENKO**

CIRCULATION
CIRCULATION MANAGER **ED UECKER**
CIRCULATION SECRETARY **JETTA ANDREONI**
CIRCULATION ASSISTANT **CHARLOTTE SEVEDGE**

EDITORIAL
EDITOR **TERRY COSTLOW**
ASSISTANT EDITOR **KATHY TEKAWA**
NORTHWESTERN REGIONAL EDITOR **ADAM OSBORNE, PhD**
SENIOR EDITOR SOUTHEASTERN REGION **BILL TURNER**
SYSTEMS EDITOR **TOM FOX**
SOFTWARE EDITOR **ALAN R. MILLER, PhD**
ASSOCIATE EDTIORS **JIM SCHREIER, HENRY DAVIS, AL BAKER**
CONTRIBUTING EDITORS **R. H. DISTLER, R. W. BEMER, ROGER C. GARRETT**

Editorial Correspondence
Direct all correspondence to the appropriate editor at: INTERFACE AGE Magazine, P.O. Box 1234, Cerritos, CA 90701.

PRODUCTION
PRODUCTION MANAGER **DAVE ANDERSON**
PRODUCTION ASSISTANT **TERRI LEDESMA**
ART DIRECTOR **FINO ORTIZ**
ARTIST **SAMANTHA LEE**
TYPOGRAPHER **MELODY A. MARTENS**

ADVERTISING
NEW ENGLAND REGION **DICK GREEN**
7 Lincoln St., Wakefield, MA 01880 (617) 245-9105

EASTERN REGION **JOHN SENSENSTEIN**
20 Community Pl., Ste. 140, Morristown, NJ 07960 (201) 267-3032

MIDWEST REGION **AL GRAVENHORST & STEVE SKINNER**
5901 N. Cicero Ave., Chicago, IL 60646 (312) 545-8621

WESTERN REGION **BRUCE BERKEY & ZACH BOVINETTE**
P.O. Box 4566, Pasadena, CA 91106 (213) 795-7002

JAPAN **TOMOYUKI INATSUKI**
Trade Media Japan Inc., R. 212 Azabu Hts., 1-5-10, Roppongi, Minato-ku, Tokyo 106
Telephone: (03) 585-0581 Telex: J28208

INTERFACE AGE EUROPE
DIRECTOR, EUROPEAN OPERATIONS **H.L. GROHMANN**
Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany
Telephone: 08106/7396

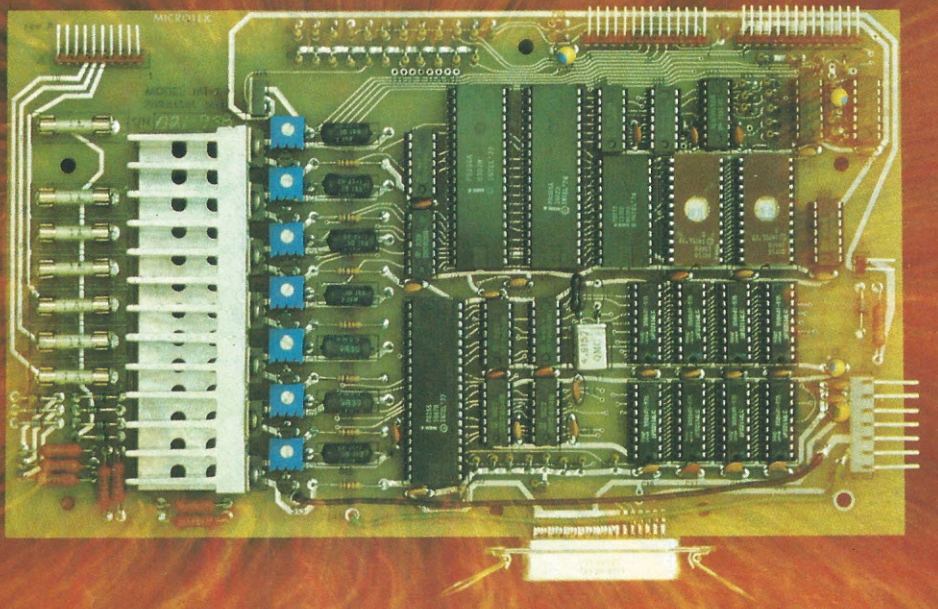
INTERNATIONAL NEWSSTAND DISTRIBUTION
LEW ULLIAN
Orberstrasse 38, D-6000 Frankfurt/M. 61, West Germany
Telephone: (0611) 44 77 90/41 84 80

INTERNATIONAL/DOMESTIC RETAIL CIRCULATION
MARY ANN LOWER (213) 926-9544

MEMBER
AMERICAN SOCIETY OF BUSINESS PRESS EDITORS

MEMBER OF THE WESTERN PUBLICATIONS ASSOCIATION wpa

INTERFACE AGE Magazine, published monthly by McPheters, Wolfe & Jones, 16704 Marquardt Ave., Cerritos, CA 90701. Subscription rates: U.S. $18.00, Canada/Mexico $20.00, all other countries $28.00. Make checks payable in U.S. funds drawn on a U.S. bank. Opinions expressed in by-lined articles do not necessarily reflect the opinion of this magazine or the publisher. Mention of products by trade name in editorial material or advertisements contained herein in no way constitutes endorsement of the product or products by this magazine or the publisher. Circulation Department, (213) 926-9540.
INTERFACE AGE Magazine COPYRIGHT © 1979 by INTERFACE AGE Inc. ALL RIGHTS RESERVED. Material in this publication may not be reproduced in any form without permission. Requests for permission should be directed to Nancy Jones, Rights and Permission, McPheters, Wolfe & Jones, 16704 Marquardt Ave., Cerritos, CA 90701.
INTERFACE AGE Magazine is catalogued in the Library of Congress, Classification No. QA75.5.155. ISSN Publication No. 0147-2992. Membership in Audit Bureau of Circulations applied for.
POSTMASTER: Please send change of address form 3579 and undelivered copies to **INTERFACE AGE** Magazine, 16704 Marquardt Ave., Cerritos, CA 90701. Second-class postage paid at Artesia, California 90701 and at additional mailing offices.



IT'S THE THOUGHT THAT COUNTS

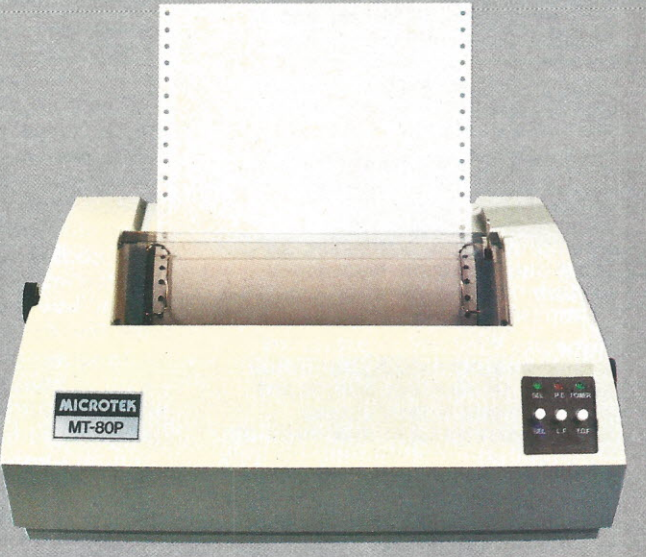
The Microtek MT-80 *looks* like a few other alphanumeric line printers on the market today. But there *is* a difference.

Our versatile, low-cost MT-80 has been designed with a more powerful brain resulting in more advanced features and more dependable performance. We believe our printer is so reliable that we offer you an incredible 365 days warranty.

If you want dependable performance, fast factory service and a low price, look for the Microtek label. The brain behind our printer really makes the difference. It's the thought that counts.

LOADED WITH INNOVATIONS

- 40, 80 or 120 columns (software selectable)
- Non-thermal paper, pin feed
- 125 CPS, 70 lines per minute
- 9 × 7 dot matrix
- Vertical format unit
- 96-character ASCII (upper and lower case)
- Adjustable forms width to 9½"
- Parallel and serial (RS-232C) interfaces available

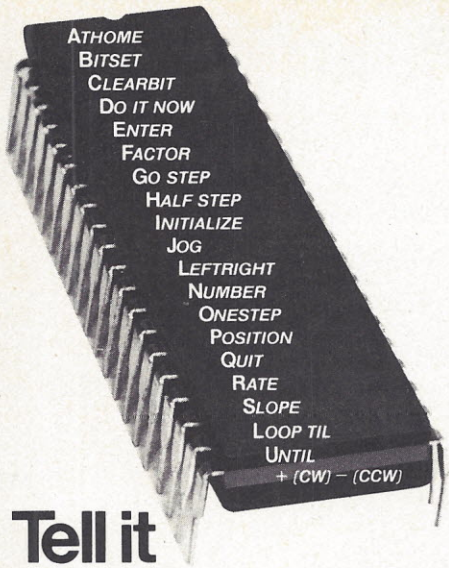


MICROTEK

For more information contact:

MICROTEK, Inc.
9514 Chesapeake Drive
San Diego, CA 92123
Tel. (714) 278-0633

CIRCLE INQUIRY NO. 51



Tell it and forget it...

. . . the new CY500 stored program stepper motor controller runs its own program, freeing your host computer for other jobs.

No more one-pulse, one-step operation requiring your host computer to tie itself down to a stepper motor. Now Cybernetic Micro Systems brings you a function-oriented stored program stepper motor controller that allows the user, or host computer, to program it and forget it . . .

The CY500 executes 22 hi-level instructions, either in command mode or as a sequence of internally-stored commands, using single byte code such as 'P' for position, 'R' for rate, and 'S' for slope. Parameter values can be expressed in ASCII-decimal for keyboard programming or binary code from the host computer. Parallel or serial communication.

The stored program capability allows the use of 'DO-WHILE' program looping and 'WAIT-UNTIL' operation. Ten different operational modes allow absolute or relative positioning, full- or half-step operation, hardware or software control of direction, start/stop, . . . and many more.

Numerous input and output control lines allow synchronizing the CY500 with external events or devices and allow each step to be triggered. Stepping at rates up to 3500 steps/sec, the CY500 also provides ramp-up, slew, and ramp-down operation, all under software control. Two interrupt lines request the host's attention if needed.

This +5 volt N-MOS TTL-compatible controller is available from stock, today, for only $95.00. Contact Cybernetic Micro Systems. We want to see you program your stepper motor and then . . . forget it.

Cybernetic Micro Systems
445-203 South San Antonio Road
Los Altos, California 94022
(415) 949-0666

(Use your MASTER CHARGE or VISA charge card)

CIRCLE INQUIRY NO. 16

EDITOR'S NOTEBOOK

As you read through the pages of this issue, notice the number of advertisers who offer hard disk storage devices with capacities of from less than 10 to over 20 million bytes of data. As the business market developed, manufacturers noted the need for a larger capacity in their data storage devices. We are just beginning to see these higher capacity devices offered by manufacturers who are designing them with the businessman in mind, from a standpoint of both convenience and price.

While the floppy disk units are beginning to show signs of maturity, what we see today with small hard disks is a young new technology with a rapid growth potential. As the hard disk technology grows the businessman will find benefits both in reduced costs and increased capacity along with more convenience, particularly in the area of removable or interchangeable media on which to store the data. The day is not far away when the term "megabyte" will become as common, if not more so, than kilobyte when referring to storage capacity for the small businessman.

Floppy disks are showing signs of advanced maturity by the decreased rate of new developments being offered. This maturity is needed by the businessman to offset the lack of advanced data handling software to go along with increased capacity for data storage.

As new products are announced at computer conventions throughout the country over the next few months, you can expect to see smaller, less expensive, larger capacity items in the hard disk area along with new operating software that will take full advantage of this improved data storage technology.

However, as they mature, these very devices in themselves will create a new area in the mass transfer or interchange of data between computers. For the immediate future, these large capacity devices relative to the floppy disk will not have a universal formatted media. That is, you can't unplug from one machine, or take the disk out of one machine and walk over and put it into another machine and use the data. How much, how fast, data will be exchanged at one time remains to be seen.

However, in the immediate future the most economic means of data transmittal is our telephone. The techniques for using the telephone to transmit data from one computer to another have been well developed. The limitations are also well known. Several companies had this in mind when they developed telephone networks for both their own private and consumer usage.

This technology has now evolved into what we call "ELECTRONIC MAIL." We have two fundamental ways of using this new technology. Currently, the most popular method is the use of a small computer to communicate via telephone with a larger system to store data. At that time or any future time, that larger system can be instructed to make the data available to other users as well.

In a few weeks we will have a system which will allow your computer to contact our computer and have programs listed in INTERFACE AGE transmitted directly into your storage device. This will save you time and effort and improve the accuracy of your copy of the program. Then, while you are in communication with the other computer, you may at your leisure review applications and new ideas that you can use on your own computer.

With the price of a complete small computer less than the cost of a terminal a few years ago, it no longer becomes economically feasible to have multiple terminals remotely communicating with a large computer. On the other hand, we have an increasing demand for an interchange of data.

Traditionally, computer and support device manufacturers hold conventions scattered across the country to announce new technology and improvements in old technology. Things to look for are announcements for smaller hard disk, hard disks with reduction in price, floppy disk, new with increased storage capacity, printers that do more and cost less, and improved computer communication devices. From the software side, we should see advancement of data storage and business programs.

All of this will be highlighted by the increased involvement of the larger companies with the home and small business computers.

I just returned from Lubbock, Texas, the home of the new Texas Instruments' 99/4 computer. My aim was to meet the people behind the development of the 99/4 and to get some hands-on experience with TI's entry into the consumer computer market, which they unveiled at the summer CES show.

To me, the most significant advancement in the home computer market is the equivalent of a Speak and Spell™ in a speech synthesizer device. What this device allows you to do is have your computer talk to you or give you instruction.

The unit, which is approximately the size of a handheld tape recorder, holds 300 words and parts of words within its own body. These 300 words are fixed and in themselves cannot be changed. In addition to these, you have a door to lift up to plug in 800 more words with a little thing about the size of a book of matches. This, of course, can be replaced with different modules, much the same as a tape cassette.

Up to this point it appears that all we have is a machine that can be programmed to say partial or complete words. The real innovation in this system lies in the TI 99/4's ability to allow you to construct complete sentences using the keyboard to store them on cassette or disk, and then to call them back during a program.

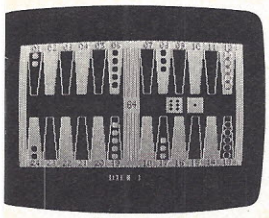
I believe we are all familiar with the original Speak and Spell™. You hit the letter "A", and the machine says "A"; or if you ask it for a spelling test it talks. However, it is limited in the number of words or vocabulary. With this new device that TI has



Even at 5:12 a.m., it's hard to quit playing Personal Software™ strategy games.

A quick game before turning in can become an all-night session when you load any of the Personal Software™ strategy games into your Apple,* PET* or TRS-80.* They'll challenge, teach and entertain you. And now there are two new games—Gammon Gambler™ and Checker King™—joining Bridge Partner,™ Time Trek™ and the best-selling Microchess.™

Gammon Gambler is a sure bet. With ten levels of skill, you can begin a novice and become an expert. Whichever level you play, the computer moves so quickly you don't have to wait. The program follows U.S. tournament rules, and includes the doubling cube to spice up the game. Written for the Apple and PET by Willy Chaplin. $19.95.



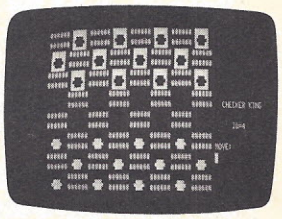
Gammon Gambler

Checker King—you probably forgot how much fun it is! If you move and change your mind, take it back and move again—without a peep from the computer. Play eight skill levels. Add and remove pieces. Save three board positions for later play. And solve three challenging checker puzzles. Written by Michael Marks for the Apple, PET and TRS-80. $19.95.

Microchess, the most widely used personal computer chess program, is a nearly perfect chess opponent for the total novice or the advanced enthusiast. Written by Peter Jennings for the Apple, PET and TRS-80. $19.95.

Bridge Partner. You against the computer in over 10 million different hands of contract bridge. You can even specify the hands' high card points. Written by George Duisman for the Apple, PET and Level II TRS-80. $19.95.

Time Trek is easy to learn, difficult to master and impossible to forget. Take command of a starship in real-time action to make the galaxy safe again. PET version by Brad Templeton. TRS-80 program by Joshua Lavinsky. $19.95.



Checker King

Personal Software, Inc., also produces the VisiCalc™ program (the software that's revolutionizing personal computing), CCA Data Management System, the Vitafacts series and other exciting software for the Apple, PET and TRS-80.

Now that you've read about the Personal Software programs, go see a demonstration. For the name of your nearest Personal Software dealer, call (408) 745-7841 or write to Personal Software, Inc., 592 Weddell Drive., Sunnyvale, CA 94086.



PERSONAL SOFTWARE

*Apple is a trademark of Apple Computer, Inc.; PET is a trademark of Commodore Business Machines, Inc.; TRS-80 is a trademark of the Radio Shack Division of Tandy Corp.

CIRCLE INQUIRY NO. 59

STRATEGY GAMES SERIES

SUPERPASCAL

5-10 times faster... and more!

Meet Pascal/Z,™ the fast, flexible compiler with higher speed, greater efficiency and improved debugging:

- **True Z-80 native code Pascal compiler**—5-10X faster than competing P-code implementations—*no interpreter required.*
- **The only multi-tasking Pascal**—produces ROMable re-entrant code.
- **Optimized** for fastest execution—recognizes and exploits special cases.
- **Easily transportable**—all hooks to your system made through support library.
- **Includes standard** floating point package. Single copy on CP/M-compatible disk includes compiler, companion macro-assembler & source of the library, $395. OEM licenses available. Write or call for more information.

InterSystems™

Ithaca Intersystems Inc., 1650 Hanshaw Road/P.O. Box 91, Ithaca, NY 14850/607-257-0190/TWX: 510255 4346

CIRCLE INQUIRY NO. 39 ©1979 Ithaca Intersystems Inc.

SWINGLINE'S DECOLLATOR SEPARATES FORMS FAST, EASY, CLEAN.

Why mess with taking apart forms? Our table-top Decollator separates carbon and carbonless continuous forms quickly and easily. Stacks 'em up nice and neat with no carbon mess. It's easy to load, no special set up or operating skills needed. There's even a variable speed control. Made in U.S.A.



Swingline 6300 Table-Top Decollator

THE Swingline COMPANY

The Swingline Company
A Division of Swingline Inc., Dept. I-4
32-00 Skillman Avenue, L.I.C., N.Y. 11101

CIRCLE INQUIRY NO. 73

announced, you can construct additional words that are not in the basic synthesizer or even in the module. You can make up your own words, using the standard built in BASIC language, storing them on disk or cassette. Later, by calling the information for playback over the speaker, you get a coherent, although admittedly one-sided, conversation from your computer.

TI makes a similar device in their minicomputer line for industrial applications. However, the synthesizer board costs more than a complete workable system in the TI 99/4.

With the money involved, I think it will be only a short time before we see this device being tucked behind a display in the store, for answering the phone and giving us current up to date vocal information. I can hear it now, "Try the canned Barlett pears on special today. Two for the price of one."

With all its market potential, TI's competition has increased, with more than one large company now entering the field of voice synthesizers or the capability for a computer to speak to you directly. We should see startling developments in this area the remainder of this year.

In the meantime, TI is expanding their software support and is actively looking for qualified companies to join them in providing software modules for their home computer. If you feel your company is qualified, and is prepared to make more than a few hundred cassettes or modules, send me a letter outlining your capabilities and I will forward it to the responsible person at TI. Write R.H. Distler, INTERFACE AGE Magazine, P.O. Box 1234, Cerritos, CA 90701.

Articles next month will be describing and showing how to use some major electronic mail systems in existence today. Also an advanced multi-microcomputer controlled printer for use a word processor printer.□

CALL FOR ARTICLES

INTERFACE AGE is seeking articles on hardware and software. Areas of interest include medical and educational applications, uses in sports, energy conservation, game playing, and small business applications.

Payment ranges from $20 to $50 per published page. Articles describing company projects or products carry the company byline, but no payment is offered. Submittals should include an abstract, outline and return envelope.

Manuscripts should be typed, double spaced with 1" margins. Minimum length is 4 pages, unless programs are included. Photos should be numbered and have a brief description attached. Tables, listings, etc. should be on separate pages. Computer listings should be printed using a new ribbon to assure good reproduction. Authors are requested to submit a statement of their background and expertise.

The publisher assumes no responsibility for artwork, photos or manuscripts. No acknowledgement is made unless accompanied with a stamped return envelope.

For submission or details contact R.H. Distler, INTERFACE AGE Magazine, 16704 Marquardt Ave., Cerritos, CA 90701. Please do not phone.

The Microcomputers you should take seriously.

The Challenger III Series is the microcomputer family with the hardware features, high level software and application programs that serious users in business and industry demand from a computer system, no matter what its size.

Since its introduction in August, 1977, the Challenger III has become one of the most successful microcomputer systems in small business, educational and industrial development applications. Tens of thousands of Challenger III's have been delivered and today hundreds of demonstrator units are set up at systems dealers around the country.

The Challenger III systems offer features which make their performance comparable with today's most powerful mini-based systems. Some of these features are:

Three processors today, more tomorrow.

The Challenger III Series is the only computer system with the three most popular processors— the 6502A, 68B00 and Z-80. This allows you to take maximum advantage of the Ohio Scientific software library and programs offered by independent suppliers and publishers. And all Challenger III's have provisions for the next generation of 16 bit micros via their 16 bit data BUS, 20 address bits, and unused processor select codes. This means you'll be able to plug a CPU expander card with two or more 16 bit micros right in to your existing Challenger III computer.

Systems Software for three processors.

Five DOS options including development, end user, and virtual data file single user systems, real time, time share, and networkable multi-user systems.

The three most popular computer languages including three types of BASIC plus FORTRAN and COBOL with more languages available from independent suppliers. And, of course, complete assembler, editor, debugger and run time packages for each of the system's microprocessors.

Applications Software for Small Business Users.

Ready made factory supported small business software including Accounts Receivable, Payables, Cash Receipts, Disbursements, General Ledger, Balance Sheet, P & L Statements, Payroll, Personnel Files, Inventory and Order Entry as stand alone packages or integrated systems. A complete word processor system with full editing and output formatting including justification, proportional spacing and hyphenation.

OS-DMS, the software star.

Ohio Scientific offers an Information Management system which provides end user intelligence far beyond what you would expect from even the most powerful mini-systems. Basically, it allows end users to store any collection of information under a Data Base Manager and then instantly obtain information, lists, reports, statistical analysis and even answers to conventional "English" questions pertinent to information in the Data Base. OS-DMS allows many applications to be computerized without any programming!

The "GT" option yields sub-microsecond microcomputing.

Ohio Scientific offers the 6502C microprocessor with 150 nanosecond main memory as the GT option on all Challenger III Series products. The system performs an average of 1.5 million instructions per second executing typical end user applications software (and that's a mix of 8, 16 and 24 bit instructions!).

Mini-system Expansion Ability.

Challenger III systems offer the greatest expansion capability in the microcomputer industry, including a full line of over 40 expansion accessories.

Networking and Distributed Processing

OS-65U level 3 now provides networking capabilities as well as time sharing ability allowing Challenger III based systems to be expanded to meet the most demanding business applications.

Prices you have to take seriously.

The Challenger III systems have phenomenal performance-to-cost ratios. The C3-S1 with 48K static RAM, dual 8" floppies, RS-232 port, BASIC and DOS has a suggested retail price of under $4000. 80 megabyte disk based systems start at under $13,000. Our OS-CP/M software package with BASIC, FORTRAN and COBOL is only $600, and other options are comparably priced.

The Challenger III Series from Ohio Scientific.

C3-B



C3-S1

C3-OEM

For literature and the name of your local dealer, CALL 1-800-321-6850 TOLL FREE.

CIRCLE INQUIRY NO. 56

C3-B wins Award of Merit at WESCON '78 as the outstanding microcomputer application for Small Business.

OHIO SCIENTIFIC
1333 SOUTH CHILLICOTHE ROAD
AURORA, OH 44202 • (216) 831-5600

This Weekend:

STIK IT.... ..to your TRS-80

That's right! Esmark's VIDIET-STIK light pen has the TRS-80 CONNECTION for LEVEL I & II. Your 4K to 48K TRS-80 System will come alive under your VIDIET-STIK within minutes of its arrival. That's because there are no wires to solder or traces to cut. You're up and running as fast as you can plug the interface into your system's cassette EAR-jack, CLOAD our custom LIGHT-WAVE demonstration software and RUN. And because the interface has a plug for your recorder, you won't have to unplug it again when loading your other software tapes. The interface allows them to pass right thru whenever you're not using the pen. It's exclusive "switched tip" design means the pen's electrically isolated from your system when it's not in use. Just point & press! It's that simple...Plug, CLOAD and RUN. And have we got the software for you to RUN with! Our demonstration tape includes a calibration program (used to adjust the CRT's brightness and contrast) plus STIK-TAC-TOE, AWARI and TOWERS. Two challenging games and a puzzle that will keep grown-ups and children Stik'ing it to your TRS-80 for hours. And there are instructions provided so you can begin writing your own light pen programs (lightware) for fun or profit (Level II). Or, just sit back and enjoy our LIGHT-WAVE tapes each month. Esmark's unmatched commitment to lightware can bring you up to five new games, puzzles, drills & educational quizzes or simulations each month. Our current LIGHT-WAVE releases are:

LIGHT-PAK 2- LIGHTPEG (4 peg-jump puzzles)
ENDRUN (Othello with a 'twist')
(LEVEL II) LIFE9 (Conway's LIFE with mutations)
Price: $19.95 (including postage & handling)

LIGHT-PAK 3- LITEGAMMON (Backgammon you'll Stik with)
(LEVEL II) STIKWUMPUS (Caves with a little 'life')
MAZEMASTER (Maze after maze to poke thru)
Price: $19.95 (including postage & handling)

Order yours now and we'll include a free copy of FLASHBACK, Esmark's newsletter dedicated to the latest news in lightware applications. And don't forget to tell your friends. The VIDIET-STIK can also be ordered for use on most other micro systems using the following processor chips:

8080 Z80 6800 6502

All that's required is a standard cassette jack leading to Ground and a readable single bit input port. Driver software is provided along with instructions for writing lightware applications. And tell your local Dealer that Esmark's got a Dealer package he won't want to miss out on. Delivery is 3 to 6 weeks from receipt of your order. C.O.D.'s are $3.00 extra but will be shipped within 2 weeks. All prices are F.O.B. Mishawaka, Indiana. Indiana residents add 4% sales tax.

ALSO COMING FROM ESMARK:

- TRS-80 Printer Interface (Cassette AUX-jack interface for all RS232 prints. Includes LLIST & LPRINT software)
- TRS-80 RS232 Communications Interface (Makes your TRS-80 a full I/O terminal to time-sharing systems the world over. Gives you intelligent or dumb terminal capabilities at 110 or 300 BAUD. Also includes Printer Interface above with 20 mA current loop & TTL level I/O options.)

—TRS-80 is a trademark of Tandy Corporation—



master charge VISA

ESMARK* INCORPORATED
507½ E. McKINLEY HWY. MISHAWAKA, IN 46544
(219) 255-3035

$62.95 PLUS $1.50 POSTAGE & HANDLING

*ELECTRONIC SYSTEMS MARKETING

CIRCLE INQUIRY NO. 27

LETTERS TO THE EDITOR

HAPPY WITH SUPERSOFT

Dear Editor:

I read about all kinds of complaints that people seem to have with both hardware and software manufacturers, but I want to relay a truly wonderful experience I had with a great software firm: Supersoft Associates.

I have a CP/M system and was looking for some good software. I saw Supersoft's ads and product announcements and decided to give them a call. I needed technical information on ENCODED/DECODE, a security system they sell for protecting data files. I asked to speak to a technical representative and was amazed to find myself connected directly with Herbert Schildt, Supersoft's Director of Software. That is real service.

He answered all my questions fully and in clear terms for me to understand. After that call I decided to buy the program and can only say that it is one of the best in my library.

If everyone in the computer business was this cooperative I'm afraid that your 'Complaint Box' would soon find itself quite empty.

Jim Weatson
Los Angeles, CA

Dear Editor:

Just wanted to say that Supersoft Associates is one great software company!

The programs are good and the customer service is simply remarkable. And my hat is off to their Director of Software Herb Schildt. He took the time to talk to me personally (something no one else has).

I hope that you print this so that others can enjoy their products. I recommend them to every CP/M user.

Allen Kune
Atlanta, GA

LOOKING FOR INFO

Dear Editor:

I am president of the Benjamin Franklin Computer Club and I am interested in any information concerning the Apple Computer system or any information on any publications put out by various organizations. Please have your readers send any information they may have.

Chris Peckham
Benjamin Franklin Computer Club
335 N. Van Dien Avenue
Ridgewood, NJ 07450

ON NUCLEAR POWER

Dear Editor:

This letter is concerned with your editorial on nuclear energy that appeared in January. I have two points that I would like to make.

First, you are the case of a non-expert telling us who knows best. While you are pointing your finger at non-experts, you are telling us what is right. Unlike Jane Fonda who is spending her own money to express her opinion, you are spending the money of the advertisers and subscribers to tell us yours. I certainly like to hear an editorial opinion but I feel it should address itself to subjects related to the publication.

Second, I disagree with your opinion. Before I go on let me explain my background. I worked for four years at Hanford, one of the three nuclear waste dumps in the United States. During this time there were official reports on leaks of hundreds of thousands of gallons of highly radioactive waste. These leaks were into the soil of eastern Washington, very close to the Columbia River. In one instance that I am personally familiar with a tank was leaking on Friday, but was not "officially" reported until Monday. This delay is not surprising considering the method used to detect leaks.

The method consisted of a truck driving around, inserting a probe into dry wells on the Hanford Nuclear Reservation and punching data on radioactivity levels into a paper tape. This tape was then brought into Richland, converted to magnetic tape and then analyzed by a program run on a CDC CYBER 74. A report was then sent back to the people responsible for waste management. If all went well and someone actually read the report then they became aware of radioactivity levels from 3 to 10 days in the past. In actuality, one tank leaked roughly 2500 gallons a day for 51 days before it was discovered. These facts were documented in the August 24, 1973 issue of *Science*.

Phil Hughes
Olympia, WA

SOFTWARE PROBLEMS

Dear Editor:

It's hard to have faith in the electronics industry when the doyen of American electronics, Texas Instruments Incorporated, is making a complete botch of its software program for the TI-59 programmable calculator, PPX.

It could be that TI is devoting so much manpower and real time to both TI-99 bugs and development of new hardware and software that the PPX program has been turned over to the lame, the halt and the blind. This is so much speculation, since that fine firm is firmly keeping its counsel about its intentions. No post cards, no phone calls; nor do I dare to call them: I picture myself indefinitely on hold while someone looks up the number of PPX in a printed directory, since TI doesn't furnish a direct line to their PPX chief.

Albert Strakna

Dear Editor:

Thank you for Carl Heintz' review of our General Ledger System in January. There were, however, a couple of points which were incorrect.

Both sides now

North Star Announces —
Double Density x 2 Sides = Quad Capacity!

The North Star Horizon now delivers quad capacity by using two-sided recording on our new mini drives! That's 360,000 bytes per diskette! A four drive North Star system accesses over 1.4 megabytes of information on-line! Think of the application flexibility that so much information storage can give you!

North Star has quadrupled the disk capacity of the Horizon computer but prices have increased a modest 15 percent. On a dollar per byte basis, that's a bargain that is hard to beat!

The proven North Star disk controller was originally designed to accommodate the two-sided drives. North Star DOS and BASIC are upgraded to handle the new capacity, yet still run existing programs with little or no change. Of course, single sided diskettes are compatible with the new disk system.

North Star Horizon Computer Prices (with 32K RAM):

Horizon-1-32K-Q	$2995
Horizon-2-32K-Q	$3595
Horizon-1-32K-D	$2695
Horizon-2-32K-D	$3095

Get both sides now! Quad capacity is available from your North Star dealer.

NorthStar

North Star Computers
1440 Fourth Street
Berkeley, CA 94710
415-527-6950 TWX/Telex 910-366-7001

CIRCLE INQUIRY NO. 55



NORTH STAR

Ace Reporter means business! This superb data base manager and report writer gives you positive control over every aspect of your business.

Accurate information, promptly available and well presented, can save you money and increase your profits.

Information about receivables, payables, sales prospects, schedules, inventory, cost tracking, estimating. Information organized the way you want it and presented the way you need it, in tables, labels, statements.

Ace Reporter helps you design a data base to your personal specification. And report formats that tell you just what you need to know. In a few minutes at your terminal even if you have absolutely no programming experience.

You can't afford not to have **Ace Reporter** working for you. It puts you in control. It gives you the edge for lower costs and higher profits.

Ace Reporter starts earning for you the day it arrives. But it won't arrive until you order it. Order today for just $240 including disk and excellent manual. (CA add $14.40 tax).

RLM Associates
1077 Ticonderoga
Sunnyvale, CA 94087
(408) 245-9441

ACE REPORTER

CIRCLE INQUIRY NO. 60

First, you were not using the current version of the General Ledger — Version 1.3, released in July, 1979 is fully menu-driven, and includes the Statement of Changes in Financial Position report.

It is quite possible to get more than nine accounts in each grouping total. All one needs to do is omit the Group Title account(s) and one may get up to ninety-one accounts which will subtotal under a Class Title account.

The effort required to make a 'post closing' entry is not really very tremendous: if one simply needs to get an amount into the Summary File, and does not need new Income or Balance sheets, the new data is entered normally; the CLOSE program will issue a message informing you that the dates aren't correct (i.e., that you've already closed), but will close the month again if you tell it to.

If you need the Income or Balance sheet to reflect the new data, all that is necessary is to rename the Summary and Detail Files, post the new data, and close again; alternately, if you have backed up your disks after each CLOSE, as is recommended, you can take a copy of last month's disk and rename this month's Detail File to current status, add the new data, and CLOSE. This latter procedure allows corrections to any earlier month (assuming you have kept backups for each month) by making the correction to the month in question, and then re-closing each succeeding month — this will be a somewhat tedious process if there are many months involved.

Peter Breeze
Structured Systems Group
Oakland, CA

Dear Mr. Breeze:

Thank you for your letter. In response:

1. *The version sent to me by your firm in August, 1979, was the version I reviewed. As a note to readers, the reason that SSG systems* could not *menu and link from one program to another is that until CBASIC (the language which SSG used) became available as CBASIC 2 recently, there was no "chain" command — i.e. no way to go from one program to another.*
2. *While it is possible to increase the number of accounts, your comment hits it right on the head — only by omitting "group title accounts" can more than nine accounts be included. While this may seem trivial to a programmer, it does make the statements look less than perfect, and requires us "beancounters" to do a little extra thinking when setting up charts of accounts.*
3. *That comment about post closing entries is good. I tried it and it works. However, it's a bit awkward if you are in the middle of a month's posting and discover you've got to do some prior period adjusting. I wish the documentation had told me about it.*

In any case, thank you for writing. We received several inquiries about SSG — all of whom wanted to know where to buy it. It is a good system.

Carl Heintz

ON THE COLUMN

Dear Editor:

February's "The Column" by Jack Purdum is the hottest item that I have encountered to date in your wonderful magazine. Thanks to Jack's genius I am now able to penetrate the mysteries of my BASIC interpreter. With his simple program (funny how great ideas can be so short and succinct) I am now able to probe around within the interpreter itself, find areas of interest and do experiments, attempt subtle modifications, etc.

In working with Jack's program, I found it useful to not exit the loops after the first discovery of a match of the requested word but instead to run through the whole interpreter to get all matches. For example the word LIST comes up three times altogether. As a matter of fact, I can't help but think that it may be a little brash to alter only the first occurrence of a word and not the others.

P.S. I just discovered that the second and third occurrences of "LIST" are part of "current program," but I still think I've got a good point.

William Smith
Berkeley, CA

Dear Editor:

Jack Purdum's column in the February issue presented an interesting technique for protecting applications software written in BASIC. However, those of us using the disk-based PolyMorphic System 8813 already have two built-in methods of protecting software.

Firstly, the disk operating system monitor (EXEC) has a special command (Setsys) which sets a flag in the disk file directory so that all programs listed in that directory appear to be "system" files — that is, files which cannot be printed, deleted, copies or otherwise tampered with. A BASIC program that has its system bit set in the file directory cannot be LISTed by the interpreter, and when stopped by the operator (Control Y) does not print the line number where the program was interrupted. BASIC data files can be similarly protected. With a machine language subroutine (which can be called from BASIC), it is possible to "toggle" the system bit of a program or data file under program control.

The second method allows the user to install an "authorization" routine which is accessed first during the system bootstrap process when the LOAD button is pushed. If the user's name is not on the authorization list and does not know his password, the user will not be able to get into the system EXEC or BASIC. Since the system EXEC



More than meets the eye.

The new Series 5000 is mighty for its size. In more than several thousand ways!

In fact, it's the first small system offering over a megabyte of integrated mini-floppy capacity. And with its super memory management, you can have better than 300k of RAM in desk or desktop versions. But hardware is just the beginning of the story.

It's the wide selection of software that really makes this system mighty.

Operating systems? Choose CP/M* with CBASIC†—the most widely accepted small computer operating system ever. Or MVT-FAMOS,** a multi-user, multi-tasking operating system with file management like the big guys, Or MICROCOBOL,†† also for multiple users, but implemented in COBOL, familiar to commercial users the world over.

And applications programs for these operating systems number in the thousands. From real estate to accounting, taxes to inventory control. they're all available at low cost—ready to run.

When you add these software and hardware features to Industrial Micro Systems' reputation for rugged, reliable quality products you'll begin to see it all. A lot more systems than your first glance reveals.

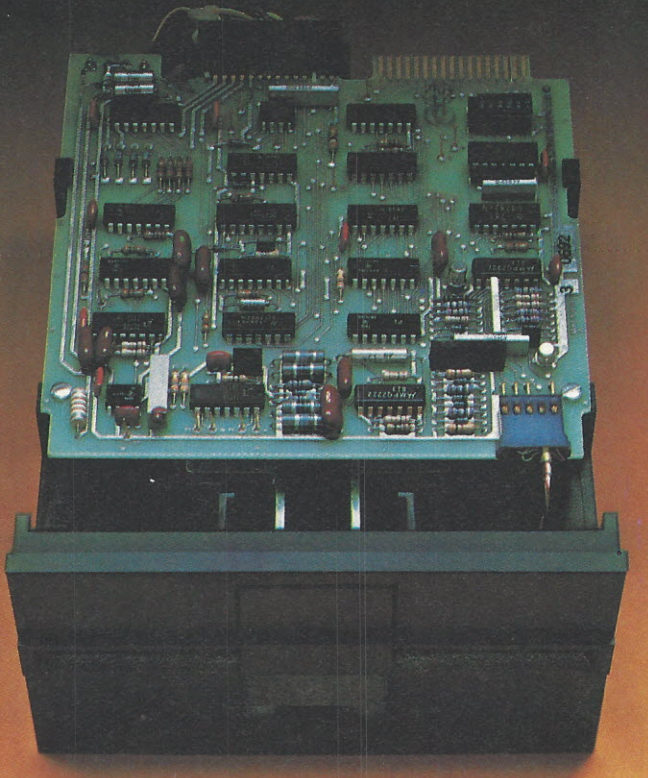
See even more at your dealer. Call us to find out the name of your nearest dealer. He'll tell you everything you need to know. And really open your eyes!

*Trademark of Digital Research Inc. **Trademark of MVT Microcomputer Systems Inc. †Trademark of Software Systems ††Product of CAP-CPP

INDUSTRIAL MICRO SYSTEMS

628 N. Eckhoff St., Orange, CA 92668, (714) 978-6966
2800 Lockheed Way, Carson City, NV 89701

If it isn't Shugart, it isn't minifloppy.™



Shugart invented the minifloppy in 1976. Today there are more than 100,000 of the little drives in use. That's because users want the affordable random access data storage of the minifloppy.

Shugart packs years of proven floppy drive technology into this tiny package. Up to 220 kbytes of data storage. Fast random access of about one-half second. And high speed data transfer of 125 kbits per second. Plus sensible, maintenance-free features like write protect to prevent accidental data loss, an activity light to indicate when the drive is selected by your computer and a door interlock to protect your media from damage.

Our proprietary read/write head provides maximum data interchange margins, and it is positioned precisely on the selected track by a patented spiral cam actuator. The DC drive motor with integral tachometer assures accurate diskette rotation and low heat dissipation. A die cast aluminum base plate provides a solid foundation for the drive.

At Shugart, technology leadership is more than a slogan, it's a commitment. Get reliability and value when you invest your money for floppy disk storage. Ask for the standard of the industry. minifloppy. If it isn't Shugart, it isn't minifloppy.

Shugart

435 Oakmead Parkway, Sunnyvale, California 94086

See opposite page for list of manufacturers featuring Shugart's minifloppy in their systems.

TM minifloppy is a registered trademark of Shugart Associates

will not load, the unauthorized user cannot use the system utilities to circumvent the authorization process.

I think that these provisions in PolyMorphic software, while not foolproof, offer the user a considerable amount of security.

David J. Barker, Ph.D.
Texas College of
Osteopathic Medicine
Fort Worth, TX

ON CP/M

Dear Editor:

I would like to respond to the letter written by Thomas L. Genung in the February 1979 issue. I feel he better stick to accounting and avoid computers and software. If he is unhappy with CP/M and its documentation, he is really going to be disappointed with other software.

In fifteen years of experience with a wide variety of hardware and software, I have encountered few off-the-shelf products that approach the level of quality that Digital Research has built into their products.

William Englander
San Diego, CA

QUESTIONING A MEMORY TEST

Dear Editor:

I am writing this letter in response to an article by Dr. Alfred Adler on "Memory Testing" which appeared in the January 1979 Micro Mathematician. The theory of memory testing described in that article is inadequate for even an elementary approach using a BASIC program. The design of programs to test memory has been an intriguing problem since the inception of electronic computers and a great deal of sophisticated statistical analysis has been required since a completely exhaustive program would take essentially forever. One, therefore, desires to use the most efficient (probability of finding memory error divided by time to execute) program possible under the given circumstances. About half of the memory defects that I have seen in the last three years of microcomputing were such that each individual bit or byte performed perfectly, but writing into some memory positions also affected others. This error can be due to defects in addressing inside or outside the chip. Any test which writes the same pattern into all the memory tested will fail to find most of these common problems which can lead to very strange behavior when executing programs (they become self-altering). Such considerations reduce the value of programs like the one presented in Dr. Adler's article.

As Dr. Adler states, the best memory test programs are written in assembly language. Simple ones can test memory without destroying its contents by complementing each byte in turn and restoring it, checking to see that both operations are carried out correctly. This still does not detect the errors described above, but it is fast and can be carried out from the monitor any time. A more sophisticated type uses a walking bit. The memory is set to all 00's and a single high bit is walked through all positions, checking that it still exists and that all other positions remain at low logic.

This test takes quite a while even in machine language and fails to find the more rare errors which are "pattern dependent." Some chips show a "spillover" of bits into their neighbors on the chip with only certain patterns of bits stored. You can see that it becomes more complicated rapidly.

However, there are several simple yet powerful memory tests which can be implemented easily in a BASIC or assembly language program. One simple test is to divide memory in half (e.g. two boards) and store some pseudo-random pattern (e.g. part of the BASIC interpreter which does not change with execution) from one board into the one to be tested. Then the patterns are compared for errors. This method has the advantage that it can be rapidly repeated (at least when using machine instructions) so that relatively low frequency intermittent failures (a common problem) may be detected. The complement of the pattern may also be used. This test is especially fast on Z-80 CPUs with block-move instructions.

A more refined variant of this test is that using a test pattern made from a known but essentially random sequence of bytes. Random number generators in BASIC interpreters often produce the same sequence of numbers when started from the same "seed." This technique can be used to create the pattern and test it without requiring division of memory. Alternately, one could use any reconstructable sequence of numbers which does not have a size equal to a power of two (e.g., most simply, counting to an odd number less than 256). Such a program would be quite efficient, being relatively thorough and fast for a BASIC memory test.

I hope this letter has given your readers some interesting and useful ideas about memory testing. Routinely checking memory is required on all computers (particularly micros without any internal error checking) and implementing a program to do it properly can be challenging and entertaining as well.

John R. Cameron, Ph.D.
Palo Alto, CA

REMARKS ON FOUNTAINHEAD

Dear Editor:

Every month I look forward to the receipt of my copy of INTERFACE AGE Magazine. I find that your magazine fulfills many informative areas that other publications do not.

In addition to your publication, I receive Datamation, Info Systems, Computerworld, Electronics and others. I believe that I do a

Continued on Page 17

Look for Shugart drives in personal computer systems made by these companies.

Altos Computer Systems
2378-B Walsh Avenue
Santa Clara, CA 95050

Apple Computer
10260 Bandley Dr.
Cupertino, CA 95014

Commodore Business Machines, Inc.
3330 Scott Boulevard
Santa Clara, CA 95050

Digital Microsystems Inc.
(Formerly Digital Systems)
4448 Piedmont Ave.
Oakland, CA 94611

Industrial Micro Systems
633 West Katella, Suite L
Orange, CA 92667

North Star Computer
2547 9th Street
Berkeley, CA 94710

Polymorphic Systems
460 Ward Dr.
Santa Barbara, CA 93111

Problem Solver Systems
20834 Lassen Street
Chatsworth, CA 91311

Processor Applications Limited
2801 E. Valley View Avenue
West Covina, CA 91792

Technico Inc.
9130 Red Branch Road
Columbia, MD 21045

Texas Electronic Instruments
5636 Etheridge
Houston, TX 77087

Thinker Toys
1201 10th Street
Berkeley, CA 94710

Shugart

Without an Eaton LRC Printer you

Get serious





are just playing computer games.

for only $389.

Introducing the New Eaton LRC 7000+ Dot-Matrix Impact Printer.

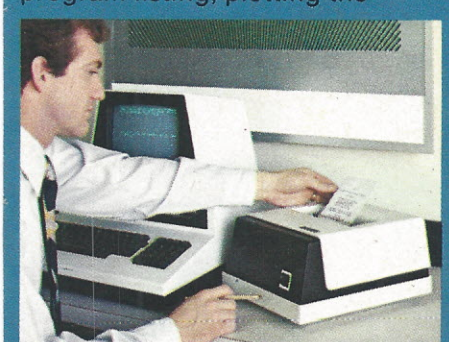
No matter what type of personal computer you now have, or are thinking of buying —

Apple	**Interact**
Cromemco	**Commodore Pet**
TRS-80	**Northstar**
TI	**Ohio Scientific**
Exidy	**and others**

— the new Eaton LRC 7000+ dot-matrix impact printer will quickly interface and be printing in a matter of seconds. You'll immediately have hard copy capabilities that will allow you to go beyond computer games and get down to serious business.

An Invaluable Addition

Adding a printer increases the practicality and usefulness of your personal computer dramatically. You'll be able to perform all sorts of bookkeeping and accounting functions — from balancing your personal checkbook to monitoring your company's inventory — and you'll have a permanent copy for your records. You'll find that whether you're controlling your family's budget, recording your program listing, plotting the



growth of a stock in the market, or any of the other thousands of analytical functions your computer can perform, a quality printer is simply invaluable. But you'll want to be sure to pick the right printer for your system.

The Right Printer



The Eaton LRC 7000+ is designed specifically for personal computers. Its plug-in simplicity makes it remarkably easy to interface with any computer. Its simple design features the fewest possible moving parts, making it virtually maintenance free. Unlike many other printers, the 7000+ can print on any type of roll paper, eliminating the hassle and added expense of purchasing a special, treated paper. Its rugged case is tough enough for industrial environments, yet attractive enough for home or office use. And most importantly, it offers the high performance and features you demand in a quality printer.

Super Performance

The 7000+ features uni-directional printing with a line speed of 1.25 lines per second. It accepts any single or two-ply paper roll from 3/4 inch to 3-7/8 inches wide, and prints a 3-1/3 inch line. Capacity is adjustable and can be 40 columns at 12 characters to the inch using the single width font; or 20 columns at 6 characters to the inch using the double-wide font. An available option allows the unit to print 64 columns at the single width setting, and 32 columns using a double width font.

An Unbeatable Price

The real beauty of the new 7000+ is the fact that you can't buy another printer that offers you all these features at anywhere near the price. Eaton LRC's 7000+ has a suggested retail price of $389.00 and comes with a 90-day factory warranty.

Specify Eaton LRC

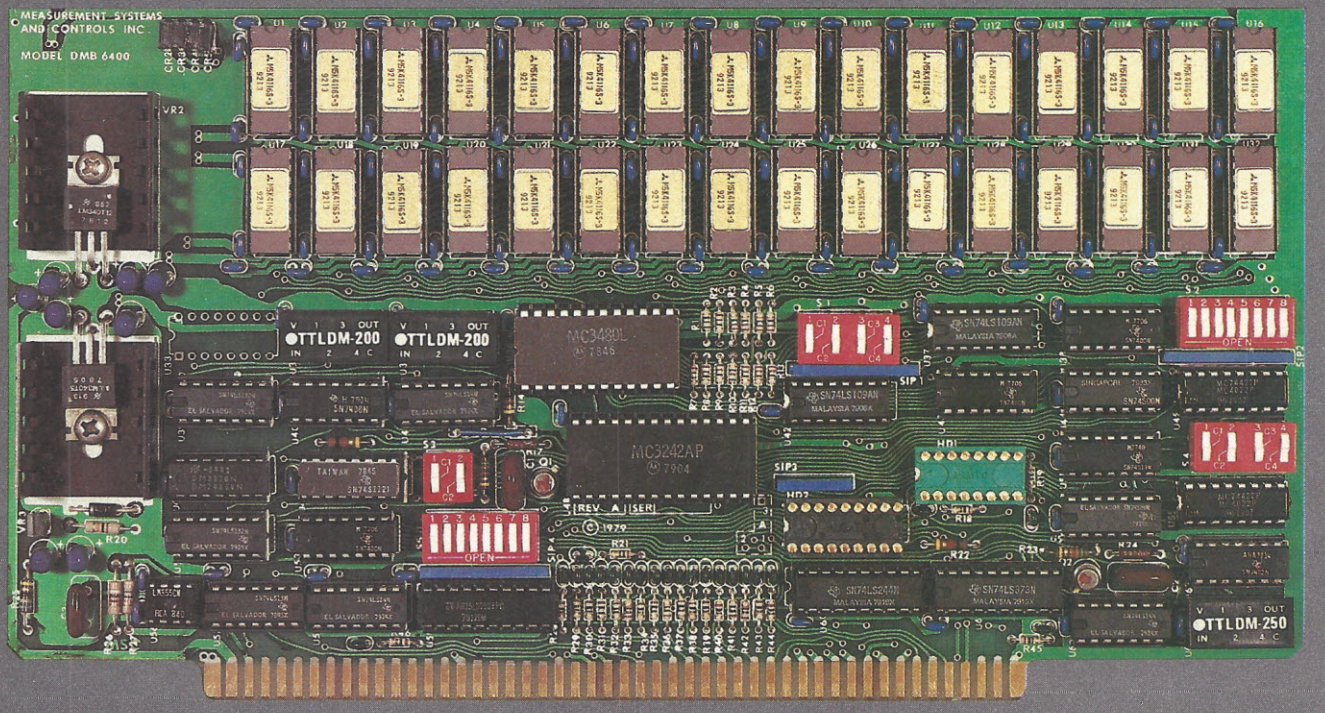
Whether you're looking for a quality printer to add to your existing computer, or are about to buy a complete system, don't settle for less than an Eaton LRC printer. Stop by your local Computerland store or contact Quest Electronics, 2322 Walsh Avenue, Santa Clara, CA 95051, (408) 988-1640 or Sigma International, Inc., P.O. Box 1118, Scottsdale, AZ 85252, (602) 994-3436 for more information. Once you've seen the new 7000+, you won't settle for anything less. LRC, an EATON company, Riverton, Wyoming 82501, (307) 856-4821.

Master distributor inquiries welcome. Contact your nearest LRC Regional Sales Office: Northeastern office, One Lakeside Office Park, Wakefield, MA 01880, (617) 245-2730; Southeastern office, 29 Delmont Drive Northeast, Atlanta, GA 30305, (404) 231-4105; Western office, 510 Lawrence Expressway, Suite 210, Sunnyvale, CA 94086, (408) 245-1590.

CIRCLE INQUIRY NO. 23

LRC

an EATON company



BANK SELECT — 64K BYTE EXPANDABLE MEMORY BOARD

MODEL DMB6400 SERIES

FULLY COMPATIBLE WITH:

ALPHA MICRO

CROMEMCO

NORTH STAR

MP/M

and most other S-100 systems

- Four independent, 16K software selectable banks.
- Switch selectable bank sizes — from 16K to 64K in 16K increments.
- Eight banks (512K) per I/O port for each of the 256 ports.
- Z-80 4MHz operation with no wait states.
- Low power — 8 watts maximum.
- Reliable, tested and burned-in memory.
- **ONE YEAR GUARANTEE**
- IEEE S-100 compatible timing.
- Attractive Dealer & OEM Prices

MEASUREMENT systems & controls incorporated

867 North Main Street ● Orange, CA 92668
Telephone: 714/633-4460

CIRCLE INQUIRY NO. 44

LETTERS
Continued from Page 13

fair job in keeping abreast of the technical world as well as the happenings within it.

One of the columns that I look to in your magazine is Adam Osborne's "From the Fountainhead." For the most part I find it informative, objective and useful.

However, when it is other than the above, it reeks of self-indulgence, unfounded rumor and gossip. It is a pity that from time to time the continuity loss can be so devastating. Specifically, and most recently, in the February issue, he launched upon what appears to be a personal attack against the Personal Computer Show run by Mr. John Dilks.

I personally have had the opportunity to visit that show for the last two years and from my point of view found it to have made significant improvements from the year before.

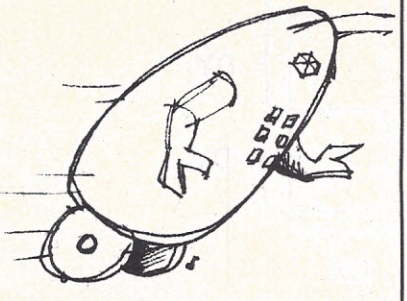
I find it hard to believe that it suffers from anything other than increased success. I sincerely wonder what Mr. Osborne's motives were. In the same column, I notice he endorsed a new entry into the computer field by a company named Microbyte. I wonder if he has really researched his endorsement fully. This industry, from my experience, has suffered from one too many overnight splashes.

No disrespect to the company, however, I would like to see some facts before I would draw a conclusion as to the success in marketing, finance, support and pedigree of the officers involved in the corporation before I would personally and professionally endorse a company no one else has heard of before.

A similarity might be drawn here for Mr. Osborne's column in the future. The Personal Computer Show that he referenced did suffer a minor drawback because of the Pope's visit to Philadelphia. It certainly causes a hysteretic affect with those trying to get to the show and avoid the monumental traffic jams.

If Mr. Osborne is trying to walk on water then someone should show him where the rocks are so he doesn't slip out of sight.

Julius Beauregard
Los Angeles, CA



The Byte Eater

HARD DISC SYSTEM



■ Complete Sub-System: DRIVE, CONTROLLER, and CABLES for S100 computers ■ Uses the most popular drive for small business systems ■ 10 Megabyte formatted capacity ■ 5MB fixed platter plus ■ 5MB 5440 removable cartridge for FAST BACK-UP and UNLIMITED off line STORAGE ■ Look Ahead buffer for maximum performance ■ expandable to 40 MB ■ Supported by CP/M 2.0, MP/M, Oasis, Famos, Fordos, Tempos Operating Systems ■ Beautiful teak finish table top (shown) enhances any office decor Rack mount also available ■ Dealers/OEMs contact John Costello.

XCOMP

XCOMP, INC., 9915-A Businesspark Ave., San Diego, CA 92131 • (714) 271-8730

CIRCLE INQUIRY NO. 82

NOW, FROM MOUNTAIN HARDWARE. THE 100,000 DAY CLOCK.™



Put your S-100 Computer on the clock.
A real time clock could double the utility of your computer. Time events in 100μS increments for up to 100,000 days (over 273 years). Program events for the same period with real time interrupts that permit pre-programmed activities to take place...without derailing on-going programs. Maintain a log of computer usage. Call up lists or appointments. Time and date printouts. Time events. An on-board battery keeps the clock running in the event of power outage.

Mountain Hardware also offers a complete line of peripheral products for many fine computers.

Available at your dealer's. Now.

Mountain Hardware, Inc.

300 Harvey West Blvd.
Santa Cruz, CA 95060 (408) 429-8600

CIRCLE INQUIRY NO. 52

WHAT'S A B TREE?

SIMPLY THE BEST APPROACH TO KEYED FILE ACCESSING!

WHY IS IT THE BEST?

A **B TREE INDEX** ensures fast insertion, retrieval and deletion of index entries, even if you have over 10,000 key values.

A **B TREE INDEX** never needs to be reorganized. It guarantees the same, short access path to every key value no matter how often the index is updated.

HOW CAN I GROW B TREES?

Easy, order **MICRO B+**™ from FairCom. **MICRO B+**™ is the first **B TREE INDEX** designed especially for microcomputers. FairCom has incorporated the most advanced features available anywhere:

- Virtual disk access to reduce index search time.
- Local node rotation to improve storage utilization.
- Linked leaf nodes to provide rapid sequential access.
- Duplicate key values accepted.

MICRO B+™ source code on disk, DEMO and Manual..$195*

MICRO B+™ Demonstration Disk and Manual....$25

MICRO B+™ source code listing...$145*

Specify **CBASIC-II** or **MICROSOFT** Basic Version 5.

Shipping $2 USA. $5 Foreign.

Disks are CP/M† compatible. 8" soft sectored format.

FAIRCOM 2606 Johnson Drive Columbia, Mo. 65201 314-445-3304

*Single CPU License †Trademark of Digital Research

CIRCLE INQUIRY NO. 29

CORRECTIONS

Due to omitted figures in "An On-Line Character Recognizer" appearing in the March 1980 issue (page 110), we are printing Figures 1 and 2 here. Some of the text is repeated for clarification.

FEATURE EXTRACTION

The feature extraction routine is called after the user has finished printing the character. The purpose of the routine is to order the data from the printed character into one or more properties which help to uniquely identify that character. Most feature extraction routines look at the areas visited by the points in the character; this algorithm is no different. Specifically, the minimum and maximum X and Y values are determined, in effect drawing a rectangle around the character. That rectangle is then divided into nine equal regions, and each point in the data collection array is "mapped" into the appropriate region, in the order in which the points were collected.

In essence, the feature extracted from the data is a map of the regions which the character visited, in the order in which they occurred. The region for any point is added to the map only if it is different from the last region to be visited. This region number can range in value from 0 through 8, and so can be represented in four bits. Two regions can thus fit in each 8-bit word, and the map for any character can fit in ten words (20 regions visited). An example of this is the letter "C" (see Figure 1). The map for this character would be 8763012, assuming the letter was drawn in a counter-clockwise fashion.

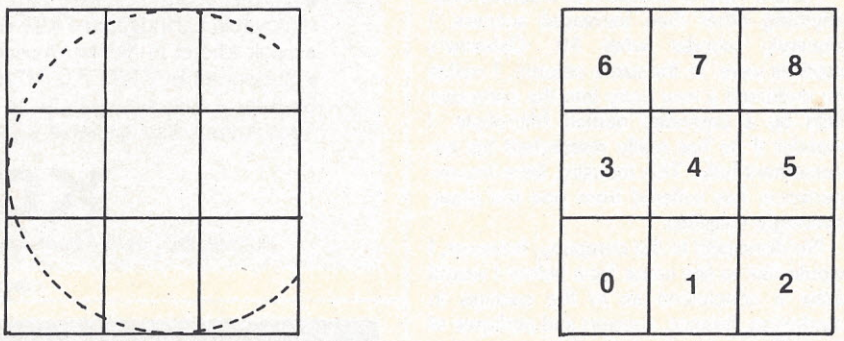


Figure 1. The letter "C" divided into 9 regions, with corresponding rectangle showing region numbering conventions.

Two additional points are worth noting. First, determining the region number for each point does require a multiply and divide capability, but a floating-point capability is not required (nor does it enhance the algorithm). The region boundary problem can be simply expressed as follows.

The respective distances between the X and Y boundary lines are

$$DX = ((XMAX - XMIN) \text{ DIV } 3) + 1$$
$$DY = ((YMAX - YMIN) \text{ DIV } 3) + 1$$

Each of the nine boxes in the rectangle measures DX by DY (see Figure 2).

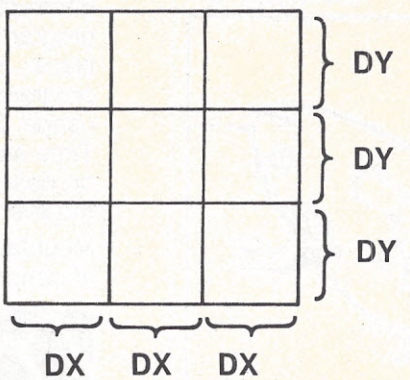


Figure 2. Rectangle divided into regions showing relationship of the value DX and DY to that rectangle.

SURVEY SHOWS KEYBOARD-TO-DISK GROWTH

According to Management Information Corporation's Seventh Annual User Survey on Data Entry Equipment, 86.8% of the configurations described by users included key-to-disk systems, while only 27.9% contained key-punch equipment. No respondents planned to purchase new key-punching equipment.

Other notable results of the survey include:

1. Service and manufacturer support of data entry equipment consistently received lower ratings than performance, reliability, and ease of use.
2. Software, interfacing, flexibility, response time and service are the most problematic areas of data entry.
3. Although 63.9% of the users felt that present data entry systems meet their needs, two-thirds of the respondents plan to purchase new data entry systems within five years, most to replace existing systems.
4. Also shown is a trend toward distributed data entry. Users are concerned with data communications and equipment interfacing capabilities which distributed data entry will require.

For more information on the User Survey, contact Management Information Corp., 140 Barclay Ctr., Cherry Hill, NJ 08034.

STUDENTS LEARN ON TRS-80

Thirty students in a Talented and Academically Gifted program at the Woodland School in Spotswood, New Jersey, are learning how to operate Radio Shack's TRS-80 microcomputer.

These students in grades four through seven have been writing their own programs covering everything from geography to graphics for about a year now. They are part of a Talented and Academically Gifted (T.A.G.) program that obtained a TRS-80 computer in January of 1979 with a grant from the New Jersey Department of Education.

At the request of teachers within the school, the students are creating and writing computer programs for use in the classroom. The programs are designed for grades one through seven in the areas of mathematics, social studies, science and language arts, and to prepare students for quizzes.

DCA CONDUCTS TESTS FOR NASA

DCA Reliability Laboratory of Sunnyvale, California, will conduct a $364,000 test, evaluation and analysis effort for NASA's Standard Parts Lead Center Office at Huntsville, Alabama. The effort will be concluded in fifteen months, after which NASA has three renewable options.

The effort will determine the reliability, physical and electrical characteristics of semiconductor components, including memories, microprocessors and peripheral chips, for use in space missions.

The effort will focus on the 1802 microprocessor and associated chip set as well as other digital, linear, discrete and passive components.

The resulting final reports will contain the methods used, summary of the findings, analysis of the statistical data and conclusions. Based on these results, the components under evaluation may be included in NASA's Standard Parts List, MIL-STD 975.

WORD PROCESSING EQUIPMENT & SERVICE FOR TVA

AM Jacquard Systems has received a $5.98 million contract to provide word processing equipment and service to the Tennessee Valley Authority, a division of AM International, Inc.

The sale, arranged by Tennessee Data Systems, a distributor for AM Jacquard Systems, includes 40 J100 minicomputers and 250 satellite terminals. The systems will be installed at 79 sites in 29 locations and serve two TVA divisons — the Office of Engineering Design and Construction and the Office of Power — in Tennessee, Alabama, Kentucky and Mississippi.

All systems have the capability of telecommunicating directly with TVA's mainframe computers. This capability provides



Cat Calls

How to tie your computer into the world.

All you need is a phone and a Cat™ acoustic modem. A Cat modem takes the data you type into your personal computer or terminal and sends it out over standard telephone lines to any other compatible computer or terminal within reach of your phone. And it listens too.

So now you can work at home and talk by phone to your office computer. Gain access to data banks. Or swap programs with computer people anywhere. The possibilities are endless—if you have Cat.

It's the fast, accurate, reliable modem that ties you into the world—for only $189.

Cat by Novation



Call for details

(800) 423-5410

In California (213) 996-5060

Available at Hamilton/Avnet, Kierulff Electronics, Byte Shops, Computerland, and your local computer store.

Novation, Inc., 18664 Oxnard Street, Tarzana, California 91356

CIRCLE INQUIRY NO. 53

word processing operators with instant transmission of text files relating to reports and operational procedures.

USE OF SPECIALIST TERMINALS IN EUROPEAN BANKS

Increased competition within the Banking and Retailing industries is likely to result in considerable changes over the next four years in the purchasing patterns of application-unique terminals for these industries. Fundamental differences in attitudes between West European countries, however, has resulted in varying national growth patterns.

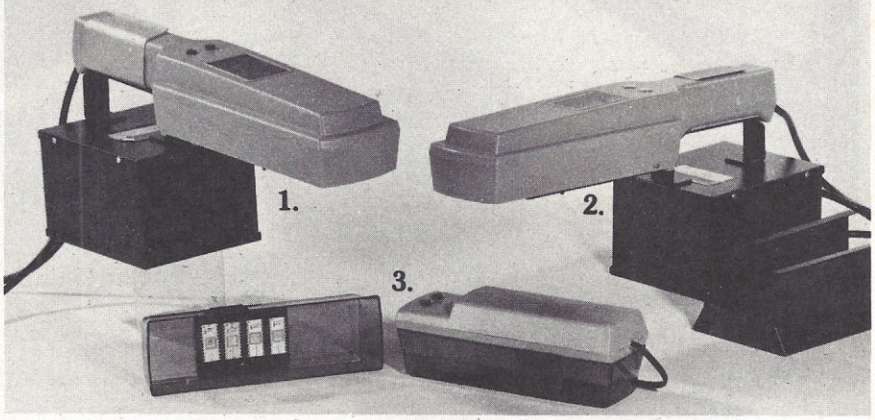
These are two of the conclusions drawn in the latest EUROCAST report from International Data Corporation: "Application Unique Terminal Markets — Western Europe 1978-1982." This study is the conclusion of a 12-month investigation of the application-unique terminal markets in Western Europe. The report also considers factory data collection, portable data entry, credit card authorization systems, and share quotation systems.

The research indicates that most major banks have already installed banking terminals; future terminal shipments for back-room usage in the branch network will be almost entirely replacement or upgrade in nature. Of the 161,900 banking terminals installed in W. Europe, IBM with 28,500 terminals, Philips (22,400), Olivetti (20,300) and Burroughs (18,100) are the market leaders.

Although consumer acceptance of automatic self-operated teller devices and cash dispensers has been slow in Europe, the market will now expand at an average annual rate of 40% over the next four years.

For more details on the EUROCAST series of reports, contact Linda Lombardi at 214 Third Ave., Waltham, MA 02254, (617) 890-3700.

FAST, DEPENDABLE EPROM ERASING



MEMORASE®
EPROM Erasing Systems

Speed, Economy, Reliability.

Engineered for all EPROM Systems Users.

1. S-52T — Fastest available EPROM eraser. Erases up to 16 chips.
2. UVS-54T — High-intensity unit with 8-chip capacity. Includes timer and tray.
3. UVS-11E — Low-cost unit erases up to four chips. Features holding tray with safety interlock system.

Applications Problem? Call Ron Cooper.

ULTRA-VIOLET PRODUCTS, INC.®
5100 Walnut Grove Ave. San Gabriel, CA 91778
(213) 285-3123

CIRCLE INQUIRY NO. 79

NATIONAL TO SECOND SOURCE MOTOROLA MACROCELL ARRAY

Motorola and National Semiconductor have signed an agreement that will allow National to manufacture and sell Bipolar VLSI circuits derived from the Motorola Macrocell Array. The terms of the agreement were not disclosed.

The Macrocell Array is a sub-nanosecond, high density (about 1200 equivalent gates), ECL Array built on Motorola's oxide isolated process. Motorola will furnish National with mask sets for the basic array, as well as technological data that will permit National to go into early production of custom VLSI circuits derived from the array.

According to Pierre Lamond, National's Vice President and Technical Director, National will go into early production on array-derived circuits for its own computer production lines, and will also entertain orders from external customers for custom circuits utilizing the array concept.

Bob Bacher, Motorola's Group Operations Manager, claims that the development of a viable second source will speed industry acceptance of the array which, he claims, is one of the most important recent developments of the Bipolar Integrated Circuit industry.

AUSTRALIAN GRAMMAR SCHOOL USES MICROCOMPUTER

Mentone Grammar School in Melbourne, Australia, is one of the first private schools in Australia to automate its business office with a microcomputer system.

With the system, the all-boy school avoids the loss of interest by billing promptly. It also permits sorting student information in any combination.

The system was developed by Data-Link, a Melbourne software program development firm, and is based on the Data General microNOVA™ microcomputer.

Although 960 boys are enrolled, the ledger contains up to 1,600 accounts owing to an overlap caused by students leaving before all accounts are paid.

A major benefit of the system is avoiding loss of interest through prompt billing and follow-up control. Through the general ledger program, Mentone Grammar now obtains printed details of individual parent's accounts and trial balances.

CALENDAR

SOFTWARE COST ESTIMATING SEMINAR

A two-day guest expert seminar dealing with the critical software cost estimating problem in government and industry will be held in two major locations: Washington, D.C. March 3-4; and Boston, MA April 7-8. The event is sponsored by the American Institute of Aeronautics and Astronautics (AIAA).

For more information, including registration procedures, contact AIAA Seminars, Dept. SCE, P.O. Box 91295, Los Angeles, CA 90009, (213) 670-2973.

OFFICE AUTOMATION CONFERENCE

The NCC Office Automation Conference will be held March 3-5 in the Georgia World Congress Center, Atlanta, Georgia.

OAC's program is designed to be a point of reference for future discussion of the industry. The conference's seminars, tutorials, panels, workshops and special events will explore the general categories of administrative management, data processing, electronic mail, micrographics, records management, reprographics, voice technology, word processing and telecommunications.

For details contact Jerry Chiffriller, 1815 N. Lynn St., Suite 805, Arlington, VA 22209, (703) 243-4100.

COMPUTER SECURITY CONFERENCE

The University of Chicago will be holding a Computer Systems and Data Security conference in San Francisco March 3-5.

Seminar attendees will learn of the many threats to computer systems and computerized information, as well as the countermeasures that can be taken.

Major topic areas include Computer Crime and Abuse, Management and Administration of the Security Function, Physical and Systems Security, Communication Security, and Data Security through Cryptography.

For details contact The Univ. of Chicago, Center for Continuing Education, MC Seminar Division, 1307 E. 60 Street, Chicago, IL 60637.

HANDS-ON WORKSHOPS

The Virginia Polytechnic Institute and State University will be holding three workshops which include Digital Electronics for Instrumentation and Automation, March 10-11; 8080-8085-Z80 Microcomputer Interfacing, Design and Software, March 12-14; and TRS-80 Interfacing and Programming for Instrumentation & Control, March 17-18.

These are hands-on workshops with the participants having the opportunity to retain the equipment.

For more information contact Dr. Linda Leffel, CEC, Virginia Tech, Blacksburg, VA 24061, (703) 961-5241.

MEDICAL COMPUTING COURSE

Human Computing Resources Corporation is offering a medical computing course entitled *The Frontiers of Medical Computing* on March 14 and 15. It explains recent advances in medical information systems in an institutional context.

Such areas as computerized patient monitoring, computer imaging, and computer-aided diagnosis will be discussed.

The course will be held at the Academy of Medicine, 288 Bloor West, Toronto. For more information contact Human Computing Resources Corp., 10 St. Mary St., Toronto, Ontario, Canada M4Y 1P9, (416) 922-1937, Mary Anne Carswell.

INTERFACE '80

Interface '80, the eighth annual Data Communications/DDP and Networking conference and exposition, will be held March 17-20 at the Miami Beach Convention Center, Miami Beach, Florida.



Got floppy disk problems?
Here's a new four letter word to use:

The word is KYBE. Because KYBE will ship you the same high performance products they've built for OEM's for years. Consistent quality media that meets the most demanding specifications. OEM's won't risk the performance of their system to less than the best media. Why risk your data either? Especially when the price is competitive.

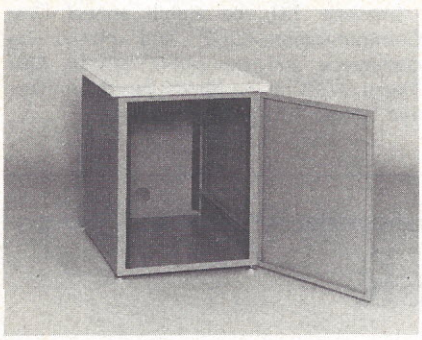
Get two day shipment on any model floppy disk, data cassette or mag card. Each is backed by an unconditional 90 day warranty and inventoried for fast delivery. Call toll free (800) 225-8715.

Dealer inquiries invited

KYBE
Dennison KYBE Corporation
132 Calvary Street, Waltham, Mass. 02154
Tel. (617) 899-0012; Telex 94-0179
Outside Mass. call toll free (800) 225-8715
Offices & representatives worldwide

DESKS AND STUFF





Computer terminals, business systems, lab components . . . they all need desks and enclosures. That's what we're all about. Computer Furniture and Accessories offers a standard line of furniture suitable for a wide variety of applications. Handsome, rugged, economical furniture in all shapes, sizes and colors. Basic models shipped from stock in days, not months. And we're nice people to deal with. What more could you ask for?

CF&A

Computer Furniture and Accessories, Inc.
1441 West 132nd Street
Gardena, CA 90249
(213) 327-7710

CIRCLE INQUIRY NO. 11

More than 250 computer industry companies will exhibit, and the conference will offer more than 60 sessions on a broad variety of topics.

For further information contact The Interface Group, 160 Speed St., Framingham, MA 01701, (800) 225-4620; in Massachusetts (617) 879-4502.

REGIONAL CONFERENCES

Expo '80, a comprehensive showcase of computer and office systems, will bring the latest in technology and information management to businesses in nine major cities.

Featured at the conference will be a wide selection of small business systems, electronic mail equipment, software products, word processing systems, general information handling equipment, CRT graphic systems and much more.

Expo '80 will start at the San Mateo, California, Fairgrounds March 19-21. Also at the Los Angeles Convention Center, March 25-27; Dallas Market Hall, April 1-3; New York Coliseum, April 22-24; Baltimore Convention Center, April 28-30; Boston Northeast Trade Center, May 6-8. The conference will appear in Detroit, Chicago and Cincinnati in June.

For more information contact The Conference Company at (800) 225-4260; in Massachusetts (617) 964-4550.

COMPUTER SCIENCE & ENGINEERING COURSES

Computer Science Education Extension will offer a series of short courses related to Computer Science and Engineering. The spring schedule consists of:

Principles of Database Systems
New York City, March 24-27
San Francisco, June 16-18

Computer Picture Processing and Graphics
New York City, June 2-4

Programming in Pascal
San Francisco, March 24-28

For more information contact Computer Science Education Extension, Computer Science Press, Inc., 9125 Fall River Lane, Potomac, MD 20854, (301) 299-2040.

SOUTHWEST COMPUTER CONFERENCE

The 1980 Southwest Computer Conference will be held in the Myriad Convention Center in Oklahoma City, Oklahoma, on March 25-27.

The conference is sponsored by Oklahoma City University in cooperation with the Association for Computing Machinery. The Conference will include 200 exhibit booths and 60 seminar presentations.

For more information contact E.Z. Million, Conference Director, Southwest Computer Conference, P.O. Box 950, Norman, OK 73070, (405) 329-3660.

INTERNATIONAL EXHIBITION AND CONFERENCE

Viewdata 80, an international exhibition and conference on viewdata, teletext and TV-based information systems, will be held in London, England, March 26-28.

The Conference has been segmented into a three-stream parallel program with both morning and afternoon sessions. Topics include: Videotex in the U.S., Canada, the U.K. and other nations; viewdata applications in travel, finance, education, the media, the office and electronic publishing; in-house videotex systems; international videotex standards; and a review of the developments on the horizon.

For more information contact Technology Marketing Analysis Corp., 680 Beach St., Suite 428, San Francisco, CA 94109.

DP TRAINING CONFERENCE

The Deltak International DP Training Conference, scheduled April 23-25, will be held at the Hyatt Regency Hotel in Chicago.

The central theme of the conference will be "1980s: The Information Decade." Speakers will include leaders in the fields of information management, data processing, telecommunications and distributed processing.

For details contact Deltak, Inc., 1220 Kensington Rd., Oak Brook, IL 60521, (312) 920-0700, Sharon Trube.

FEDERAL DP EXPO

The sixth annual conference and exposition addressing computer system users in the U.S. government will be held April 28-30 at the Sheraton Washington Hotel in Washington, D.C.

More than 200 computer industry companies will display and demonstrate hardware/software products and services.

The conference portion will feature talks on a broad variety of topics by leading ADP and Datacomm experts.

For details contact The Interface Group, 160 Speen St., Framingham, MA 01701, (800) 225-4620; in Massachusetts (617) 879-4502.

MICRO EXPO '80

The fifth Micro Expo '80 will be held May 6-8 at the Centre International de Paris in Paris, France. Micro Expo '80 is the largest show in Europe on microcomputers and personal computing.

The four major themes of this year's technical program are Personal Computing, New Products, Industrial Applications and Communications.

For details contact Robert Maraziti, U.S.A. Show Coordinator, Sybex, Inc., 2344 Sixth St., Berkeley, CA 94710, (415) 848-8233.



When your Data Demands Security.

ENCODE/DECODE: *THE* Software Security System for CP/M* & TRS-80# DOS

Let's face it, there is information which just isn't meant for everyone who uses or has access to your computer. Consider payroll or tax records. Until now, the only way to secure these and other valued or privileged records meant either "pulling the plug" or locking the discettes in a safe. Who wants to run to the safe each time an update needs to be made? At last a simple, effective and convenient method of data security is available—ENCODE/DECODE.

ENCODE/DECODE is a complete software security system for your micro/mini computer. ENCODE/DECODE can provide both the level of security and privacy you desire without loss of ON-LINE immediate access to data. ENCODE/DECODE is a sophisticated coding program which transforms data stored on disc into coded text which is completely unrecognizable. When it's time to access the file, it is decoded and ready for use. This means that data can be on-line and current with all your other files, yet only the user defined combination can retrieve it.

Multiple Security levels: Using ENCODE/DECODE you can easily maintain several layers of security through the use of separate combinations. This means that each file can have its own 'password' allowing only those with the 'password' access to the file.

ENCODE/DECODE uses a complex coding algorithm which supports over 987,000,000 possible combinations thus making accidental or 'exhaustive search' methods of decoding virtually impossible. Briefly, an encoded data file will appear scrambled and completely unintelligible until you decode it. Both encoding and decoding require the user defined combination.

Uses for ENCODE/DECODE are unlimited. Below are a few examples:

data bases	general ledger	inventory
payroll files	correspondence	accounts payable/ receivable
programs	tax records	& more
text	mail lists	

ENCODE/DECODE is available in two versions. ENCODE/DECODE I provides a level of security suitable for normal use. ENCODE/DECODE II provides enhanced security for the most demanding needs. Both versions come supplied on discette and with a complete user's manual.

ENCODE/DECODE I : $ 50.00
ENCODE/DECODE II: $100.00
manual for above: $ 15.00

Minimal system requirements: 24K CP/M; 16K disc for TRS-80

formats: CP/M 8" SOFT SECTORED, NORTHSTAR CP/M AND TRS-80 DOS

All Orders and General Information:
SUPERSOFT ASSOCIATES
P.O. BOX 1628
CHAMPAIGN, IL 61820
(217) 344-7596

Technical Hot Line: (217) 384-0847
(answered only when technician is available)

OEM and dealer inquiries invited, overseas orders add $5.00 shipping.

*CP/M REGISTERED TRADEMARK DIGITAL RESEARCH
#TRS-80 TRADEMARK TANDY CORP

TRS-80® CP/M® & CBASIC® BUSINESS SOFTWARE

All MOD II CP/M's are ***not*** created equal . . .
Find out why ours is the **Better Business Buy!**

Model I CP/M (rel. 1.5) $150.00
Model II CP/M (rel. 2.0) 250.00
CBASIC 95.00

APH (Automated Patient History) 175.00
RESIDENTIAL PROPERTY ANALYSIS system 300.00

The Genuine Article:
Osborne & Assoc. CBASIC source programs—
O&A **Payroll** w/Cost Accounting $250.00
O&A **Accts. Rec./Accts. Payable** . . . 250.00
O&A **General Ledger** w/Cash Journal . 250.00
O&A CBASIC books for above (each) 20.00

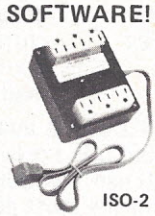
Send 30¢ SASE for CP/M Users Group software list & free "CP/M Primer".

® TRS-80 is a registered trademark of Radio Shack, a Tandy company
* CP/M is a registered trademark of Digital Research
® CBASIC is a registered trademark of Software Systems

CYBERNETICS INC (714) 848-1922
8041 NEWMAN AVENUE • SUITE 208 • HUNTINGTON BEACH, CALIFORNIA 92647

**DISK DRIVE WOES? PRINTER INTERACTION?
MEMORY LOSS? ERRATIC OPERATION?
DON'T BLAME THE SOFTWARE!**


ISO-1


ISO-2

Power Line Spikes, Surges & Hash could be the culprit!
Floppies, printers, memory & processor often interact!
Our unique ISOLATORS eliminate equipment interaction AND curb damaging Power Line Spikes, Surges and Hash.

***ISOLATOR (ISO-1A) 3 filter isolated 3-prong sockets; integral Surge/Spike Suppression; 1875 W Maximum load, 1 KW load any socket $54.95**
***ISOLATOR (ISO-2) 2 filter isolated 3-prong socket banks; (6 sockets total); integral Spike/Surge Suppression; 1875 W Max load, 1 KW either bank $54.95**
***SUPER ISOLATOR (ISO-3), similar to ISO-1A except double filtering & Suppression $79.95**
***ISOLATOR (ISO-4), similar to ISO-1A except unit has 6 individually filtered sockets $93.95**
***ISOLATOR (ISO-5), similar to ISO-2 except unit has 3 socket banks, 9 sockets total . . . $76.95**
***CIRCUIT BREAKER, any model (add-CB) Add $ 6.00**
***CKT BRKR/SWITCH/PILOT any model (-CBS) Add $11.00**

PHONE ORDERS 1-617-655-1532
ESP Electronic Specialists, Inc.
171 South Main Street, Natick, Mass. 01760
Dept. IF

CIRCLE INQUIRY NO. 26

the Column

By Leon Felkins

BINARY ARITHMETIC USING THE HANDHELD CALCULATOR

INTRODUCTION

There are advantages in being able to do binary (base 2) arithmetic on a calculator. Or at least, check answers on a calculator while learning base 2 arithmetic. If the occasion arises to do base 2 arithmetic, two approaches could be taken: (1) The arithmetic would actually be figured in base 2 or (2) numbers would be converted to base 10, the calculation figured, then converted back to base 2.

Both methods have a few serious drawbacks. With method 1, the primary problem is that it cannot be done on a calculator except on the special purpose calculator for Binary operations or a programmable version with the appropriate program. Of course, it can be done manually. However, it is not so easy and is very susceptible to errors.

The main problem with method 2 is the time it takes. Obviously, on a programmable calculator a program can be implemented that will convert from base 2 to base 10, do the operation and convert back to base 2. (The program is not as simple as one might think, and takes quite a few steps.) But when the program is used, doing a simple base 2 addition seems to take hours.

This article proposes a third method that eliminates the above objections. Base 2 calculations can be done on a standard base 10 calculator without a program. It is as fast as base 10 calculations.

BASE 2 ADDITION

The method used in implementing BCD arithmetic on the computer provides the nucleus of an idea for doing base 2 calculations in base 10. Full details on BCD arithmetic will not be given here; the interested reader can find details in most any basic text on digital theory (see references). However, the similarity to our problem will be pointed out. In BCD arithmetic, only the digits 0 to 9 are used of the possible 16 hexadecimal digits. However, when an addition of two digits is performed, the answer may be one of the forbidden digits. If this happens, adding 6 will give the correct result.

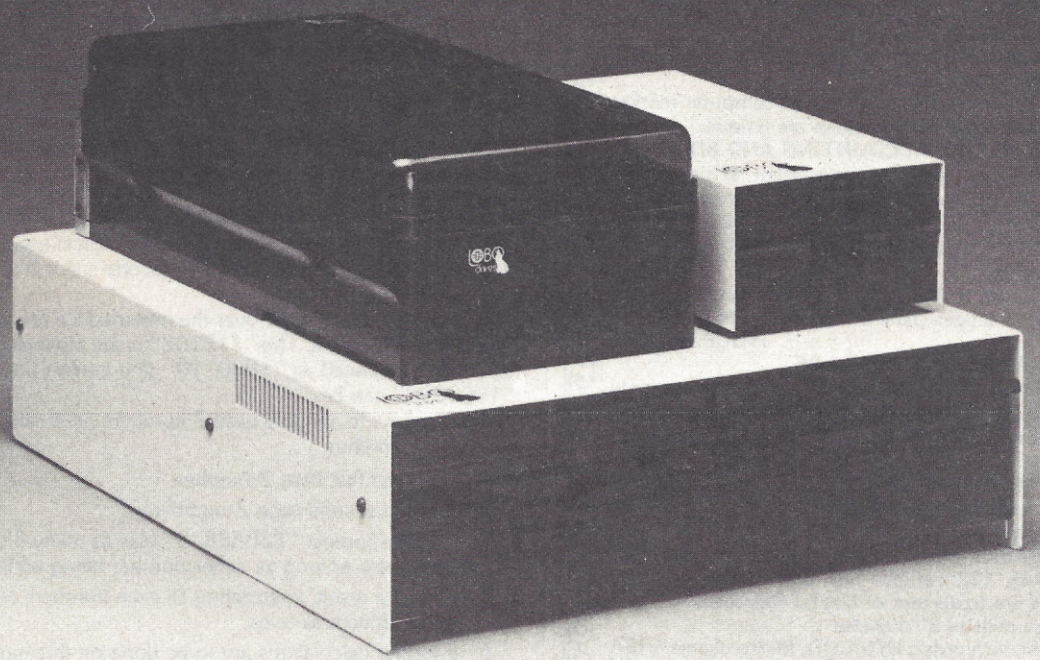
Since in BCD arithmetic the problem of obtaining base 10 answers while working in base 16 is solved by adding 6, one wonders if a similar method might work in doing base 2 arithmetic on a base 10 calculator. The answer is, fortunately, yes, and the number to add is 8. It should be pointed out, however, that the BCD calculations require a rather complex algorithm to add "6" only when needed. The same problem in our base 2 method might be expected.

Then, it appears that base 2 addition can be done if 8's could be selectively added as needed. Look at all possible cases.

Case 1	Case 2	Case 3	Case 4
0	0	1	1
+0	+1	+0	+ 1
0	1	1	2
			+ 8
			10
	No "8" needed		"8" added

It turns out that, even though in only one case in four does 8 need to be added, the simplest method is to always add 8 and to subtract it out again when not needed (again, this comes from the BCD addition algorithm).

NEW FROM LOBO:

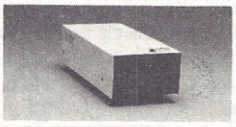


An Entire Family of Disk Drives for APPLE, TRS-80*, and S-100 Computers

Only LOBO DRIVES offers you an entire family of fully-compatible disk drives to select from. Whatever computer you're using, APPLE, TRS-80, or S-100, you can add a LOBO drive now, with the peace-of-mind of knowing there's a whole family of drives available when you're ready to expand.

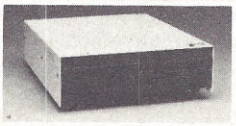
And every drive you order comes complete with chassis and high reliability power supply. Each drive is 100% calibrated, burned-in, and performance tested on either an APPLE, TRS-80, or S-100 computer before it's shipped. We are so proud of our drives . . . our quality, reliability, and performance, that we back-up every drive with a one year, 100% parts/labor warranty.

400 SERIES FLOPPY DISK DRIVES



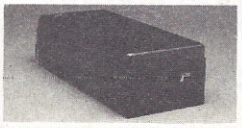
Meet our low-cost 5.25-inch mini drive that records data in either hard or soft sectored format. It is available in single or double density configurations, with a total storage capacity of 220K bytes.

800/801 SERIES FLOPPY DISK DRIVES



Here is our dual 8-inch Floppy disk memory unit. It records and retrieves data on standard 8-inch diskettes to provide 800K bytes of data storage unformatted, or 512K bytes in IBM format per drive. It is also available with double-sided, double-density capabilities, for a maximum storage capacity of 1.6 Megabytes.

7000 SERIES HARD DISK DRIVES



The latest member of our drive family, the Series 7000 is an 8-inch, 10 Megabyte Winchester Technology, hard disk drive. It is fully hardware/software compatible and comes complete with disk controller. Now you can have the convenience, speed, reliability, and all the storage capacity you need.

Call or write for the complete LOBO DRIVES story. Find out just how competitively priced a quality drive can be.

Quantity discounts available –
Dealer inquiries invited.

Yes, I want to know more about LOBO Drives and what they can do. Send me information on:
☐ TRS-80 ☐ APPLE ☐ S-100

☐ 5 1/4-in. Floppy drive
☐ 8-in. Floppy drive
Single sided
Double sided
☐ 8-in. Winchester hard disk, 10 Mbyte drive
☐ Double density expansion interface

Name _____
Company _____
Address _____
City _____ State _____ Zip _____
Phone No. _____
If dealer, provide resale no. _____

LOBO drives INTERNATIONAL

935 Camino Del Sur
Goleta, California 93017
(805) 685-4546

"CAN YOU REALLY AFFORD TO PAY LESS?"

*TRS-80 is a registered trademark of Radio Shack, a Tandy Company.

Put Your Micro To Work

— with packages from ASYST Design Services

With four years experience in minicomputer applications, we've demanded a lot from packages.

Like high standards in programming, documentation and ease of operation.

We've brought this expertise to the microcomputer market starting with three packages we believe are unique.

- **PROFESSIONAL TIME ACCOUNTING AND BILLING**
 For the accountant, attorney, consultant. If time is the basis of your business, this package will enable you to keep track of it. **PRICE: $595.00**
- **APPLICATION DEVELOPMENT UTILITIES AND TEMPLATES**
 Programs to help you get applications running in the shortest time. Includes General Subroutine Package. **PRICE: $495.00**
- **GENERAL SUBROUTINE PACKAGE**
 More than thirty subroutines to perform commonly required functions. Includes screen, data, index file management and report formatting. **PRICE: $295.00**

Also available:

MICRO-AP SELECTOR III–C2 $345.00
OSBORNE ASSOCIATES PAYROLL $250.00

Operating environment:
CP/M™ CBASIC–2 system with dual drive, minimum 48K of RAM, 132 column line printer and 24 x 80 CRT. (CP/M is a trademark of Digital Research)

Shipping: Single density 8" diskette

Terms: Payment with order, UPS/COD, Mastercharge/VISA

ASYST Design Services
756 Bowling Green
Cortland, N.Y. 13045
(607) 756-8247

CIRCLE INQUIRY NO. 5

THE BEST
MICRO COMPUTER
– Exporter –
"That's Right"
Σ Sigma INTERNATIONAL INC.
Fast + Efficient

Our Foreign Customers Receive:
- Purchasing Agent's Service
- Wholesale Discounts
- 220 V/50 Hz Power

Products

NORTH STAR	TELEVIDEO
TEXAS INSTRUMENTS	BASE 2
SOROC	KONAN CORPORATION
INDUSTRIAL MICRO	CENTRONICS
HAZELTINE	INTEGRAL DATA
EXIDY SORCERER	VERBATIM
MICRO PERIPHERALS	**+ More**

P.O. Box 1118 Scottsdale, Arizona 85252 U.S.A.
Tel: 602-994-3435 TELEX: 165745

CIRCLE INQUIRY NO. 65

Try an example.

```
   101101
    11001
  -------
   112102   Calculator result
  +888888   Add 8's
  -------
  1000990
-     880
Selectively subtract 8's
  -------
  1000110   Correct result
```

The operation of "selectively subtracting 8's from digits greater than 1" is fairly difficult to implement on a programmable calculator, but easy to do manually. As a matter of practical use, just read even numbers as zero's and odd numbers as one's. (Or 8's as 0's and 9's as 1's.)

This is an improvement over the intermediate result obtained before adding the eights. The "112102" result obtained above cannot be easily interpreted as "1000110" (the correct answer), whereas "1000990" can be.

In summary, to add two base 2 numbers on a calculator, use the following algorithm:

1. Enter the first base 2 number.
2. Add the second base 2 number.
3. Add the number "888888--8" (use as many 8's as needed to be at least as long as either number being added).
4. Read the result, interpreting all even numbers as zeros and all odd numbers as ones.
5. If further calculations are to be done on this number, convert it to all ones and zeros by selectively changing odd digits to ones and even digits to zeros. (Easily done by subtracting 8's where needed.)

The operations will be more convenient if a "row of eights" is first stored to the limit of the calculator, in memory. One caution: the method only works with two numbers at a time.

SUBTRACTION

Surprisingly, subtraction turns out to be simpler than addition. Simply subtract one binary number from the other, interpreting the results by step 4 above.

Example:

```
  1001010
 -110111
 -------
   890899
```

Subtraction works regardless of the relative size of the two numbers, always giving the correct signed result (subject to the interpretation of the 8's and 9's by step 4 of the Addition Algorithm).

MULTIPLICATION

Since it was pointed out that the addition technique only works for two numbers at a time and since multiplication effectively requires the addition of a column of numbers, it might appear that developing a simple multiplication algorithm would cause trouble. Fortunately, a simple method is feasible, as stated in the following algorithm.

The multiplication algorithm:

1. Enter the first number as written. Enter the second number, converting all 1's to 9's except the last one.
2. Multiply as usual.
3. The result is obtained by interpreting all 9's and 1's as 1's and all 8's and 0's as 0's, as in addition and subtraction.
4. If any threes or sevens appear in the result, reverse the order of multiplication and enter again.

A few examples:

$110111 \times 1011 = ?$ $(55 \times 11 = 605$ base 10)

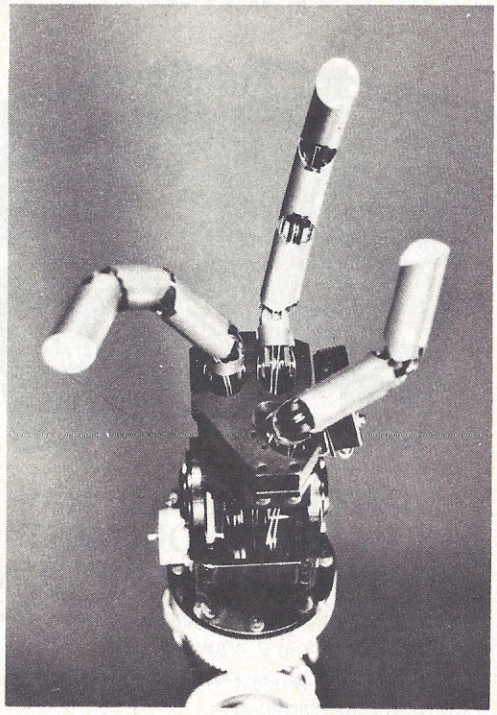
Enter as:

$110111 \times 9091 =$ Result: 1001019101

Interpret as 1001011101, which is the correct result.

$101101 \times 10101 = ?$

GIVE YOUR COMPUTER A HAND



Have you ever wanted to do more with your micro than play computer games and balance your checkbook?

Finally, there is a unique new quarterly magazine that gives you all the information you need to transform your home computer into a working *ROBOT!*

ROBOTICS AGE covers every aspect of robot research and experimentation—from the basic principles to the latest developments in laboratories around the world. In our features, special emphasis is given to plans, circuits, and programs that you can use in your own microcomputer-controlled robot. Each article is designed to be understandable to the novice experimenter, but with technical detail and complete references that will satisfy even the professional researcher. Added to that are robotics-related New Products, Book Reviews, abstracts of selected recent technical papers, and reports on how you can participate in the growing number of robotics and Artificial Intelligence organizations in the US and abroad.

The explosive growth in the use of robots has been described as the Third Industrial Revolution. Robots are having a growing impact on the shape of industrial society—and will be a significant part of your own future. Join the thousands of ROBOTICS AGE readers already gaining a deeper understanding of this fascinating and significant science. Learn how you can contribute to the development of the intelligent robots of the future—*Subscribe Today!*

ROBOTICS AGE



ROBOTICS AGE P. O. BOX 801 LA CANADA, CA 91011

☐ 1 year subscription (4 issues)—US: $8.50; Canada & Mexico: $10.00; Foreign: $12.00†

☐ 2 years (8 issues) US: $16.00; Canada & Mexico: $19.00; Foreign: $23.00†

Payment enclosed ☐ Bill my VISA ☐ MasterCharge ☐ Bill me (N. America only) ☐

Number Expires Signature (Required)

Name (Please Print)

Company/Title

Street Address/Apt. No.

City State/Country Postal Code

† Rates shown are for surface delivery. Air mail delivery available at extra cost. Please remit in US funds drawn on a US bank.

YOUR INVITATION TO COMPUTING ADVENTURE



INTERFACE AGE HAS EVERYTHING YOU WANT TO KNOW ABOUT COMPUTING

Every issue is written and edited by experts. Every issue brings you how-to's and know-how on:

- Tutorials on all phases of computing technology
- Software and Hardware
- New Product Design and Reviews
- Business, Accounting, Investment Analysis, Law and Medical Applications
- Educational and Developmental Applications
- Personal and Home Projects
- Games, Robotics and Energy

It's a 12-month program to help you enjoy all computing activities and it's all yours. All the spectacular computing fun, adventure and know-how that you have dreamed about. We guarantee* it.

DON'T WAIT! SUBSCRIBE TODAY!

***Unless you are completely satisfied with INTERFACE AGE, you may cancel your subscription at any time and receive a full refund for all unmailed copies.**

Please enter my subscription for:

☐ 1 Year (12 Issues) $18
☐ 1 Year Canada/Mexico $20
☐ 1 Year Foreign Surface $28 ☐ Air Mail $50

Check or Money Order must be in U.S. Funds drawn on U.S. bank.

Make check or money order payable to:
INTERFACE AGE Magazine, P.O. Box 1234, Cerritos, CA 90701

☐ Bill Me Please charge my ☐ AMERICAN EXPRESS Cards Welcome ☐ master charge ☐ VISA

Signature _____ Expires

RUSH my copies to:

First Initial		Middle Initial		Surname	
Mail To:					
City			State	Zip Code	

Enter as:

101101 × 90901 = Result: 9190182001

bad digit

Re-enter as: 10101 × 909901 = ?

The result is 9190910009, which when interpreted as 1110110001, is the correct result.

Could fractional numbers be multiplied by this method? Actually, there is no real difference to the calculator and it does, in fact, work. The only precaution is to refrain from entering numbers that exceed the range of the calculator.

An example:

1.01 × 111.101 = ?

Enter as:

1.01 × 999.901 = Result: 1009.90009

Interpret as 1001.10001.

It is possible in rare combinations of larger numbers to get a 2 or 7 in the result, no matter in which order the numbers are multiplied. Since the calculator is limited to small numbers anyway, this is not a serious problem.

DIVISION

As one might agree, it would not be reasonable to expect division to work. Since internally the calculator works in "floating point," it cannot be expected to get all 0's, 1's, 8's or 9's when doing a division operation.

However, it does work well enough to be useful. In some cases it will give exactly the right answer; in other cases, it requires some imaginative interpretation.

Try a few examples. The binary numbers are entered unmodified.

Example 1: (Even divisor — no remainder)

In base 10: 230/10 = 23
In binary: 11100110/1010 = 10111

On the calculator:
Enter 11100110/1010 = ?
Result is 10990.20792; close but not exact.

Example 2:

In base 10: 120/5 = 24
Enter in binary on calculator as
11110000/101 = ?
Result is 11000, which is correct.

How about a non zero remainder problem?

Example 3:

Base 10: 10/3 = 3.3333
Enter in binary on calculator as
1010/11 = ?
Result is 91.818181
which is correct when properly interpreted.

WHY IT WORKS (MOST OF THE TIME)

When a 2 or 3 turns up in the result of an addition, it should be converted to 10 or 11 to get the correct binary result. Obviously, adding an 8 will do the trick. A detailed analysis of the multiplication of numbers up to 6 digits long will show that the necessary carry will always be generated (by the algorithm stated above) except in a few rare cases. The addition of special rules to handle these few cases would cause more confusion than it would be worth.

SUMMARY

This article has shown that with a few simple rules, the handheld calculator can be used to simply and efficiently perform Base 2 arithmetic. While addition and subtraction always give the correct results, multiplication and division do not always give the exact answer. However, the results are correct enough to make the use of these algorithms very worthwhile.☐

At Last! Affordable
WORD PROCESSING
Made Possible by TRS-80™

NEW SCRIPSIT™ Word Processing Software for 16K Level II TRS-80s.
$69.95 Cassette*
$99.95 Disk*

EDIT, DELETE, MODIFY— then print it error-free, 45 characters per second!



NEW UPPER AND LOWER CASE KIT. Available for new or existing systems.
$99 Installed*

NEW TRS-80 DAISY WHEEL PRINTER. Clean and readable printing, like the very finest electric typewriters.
$2999*

RADIO SHACK BRINGS YOU A COMPLETE WORD PROCESSING SYSTEM FOR TRS-80 MODEL I COMPUTERS.

Radio Shack smashes another computer "cost barrier" with the new TRS-80 Word Processing System. The system includes our new SCRIPSIT software, Upper/Lower Case Kit and Daisy Wheel Printer. Add it to any 16K Level II TRS-80, or buy a complete system. Once you've tried it, you may never want to use a typewriter again!

The new SCRIPSIT software lets you compose letters and documents of all types on TRS-80's screen in upper case, or upper and lower case with the new Upper/Lower Case Kit. You can move words or entire paragraphs, insert, delete and edit to your heart's content! SCRIPSIT gives you automatic page numbers, page headings and footnotes and makes it easy to indent paragraphs, change line widths, and center your text horizontally or vertically. Advanced features include justification, hyphenation, global search/replace, and variable screen width. On-going reports, form letters and text with print commands can be stored on TRS-80 cassettes or diskettes for use or revision at any time.



SCRIPSIT software includes an audio cassette course that makes anyone a proficient word processing operator.

PRINT ALL OF THE "ORIGINALS" YOU NEED, FAST AND ERROR-FREE!

Our new WP-50 Daisy Wheel Printer is fast and gives you the same quality of the finest electric typewriters — carbon film ribbon and all! Or, if your job doesn't require "letter" quality, a TRS-80 system with a dot matrix, u/lc printer costs even less.

A complete TRS-80 cassette system with Word Processing Software, Upper/Lower Case Kit and a dot matrix printer is yours for just $2,046.95*. Or choose a really deluxe system with the WP-50 Printer and two floppy disks that store eight hours of 50 WPM typing for only $5,492.95*.

Sound exciting? You bet it is! Visit your nearest Radio Shack outlet or write for details.

*Retail prices may vary at individual stores and dealers.

Mail to: Radio Shack, Dept. CMA-429
1300 One Tandy Center
Fort Worth, Texas 76102

I'd Like to Know More!

☐ **Send details on TRS-80 Word Processing and the 24-page TRS-80 Catalog #RSC-3.**

☐ **Have a representative contact me.**

NAME _____

ADDRESS _____

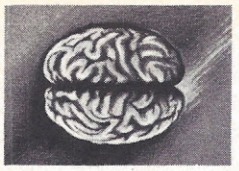
CITY _____

STATE _____ **ZIP** _____

I Own/Use a TRS-80 ☐ **Yes** ☐ **No**

Model _____

Radio Shack®
The biggest name in little computers™
A DIVISION OF TANDY CORPORATION • FORT WORTH, TEXAS 76102



8086 Boards

CPU with Vectored Interrupts **$650.**

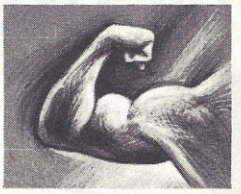
PROM-I/O **$495.**

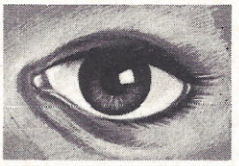
RAM **$395.**
8K x 16/16K x 8

ANALOG Boards

A/D 16 Channel, **$495.**
12 Bit, High Speed

D/A 4 Channel, **$395.**
12 Bit, High Speed





VIDEO DIGITIZATION

Real Time Video **$850.**
Digitizer and Display

Computer Portrait System **$4950.**

S-100 Boards

Video and/or Analog
Data Acquisition
Microcomputer Systems

TECMAR INC.

The High Performance S-100 People

TECMAR, INC.
23414 Greenlawn • Cleveland,OH 44122
(216) 382-7599

CIRCLE INQUIRY NO. 80

Tarbell Double Density Floppy Disk Interface

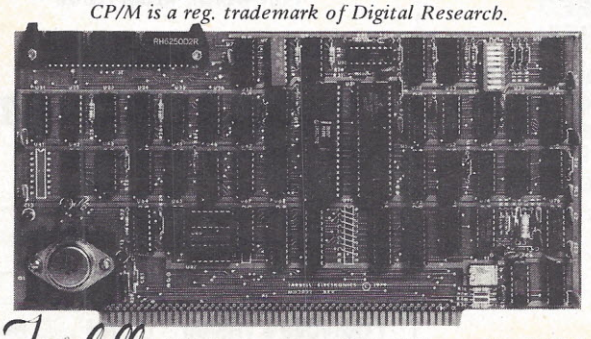
FOR 8" DISK DRIVES

Under Tarbell Double-Density CP/M, single and double density disks may be intermixed. The system automatically determines whether single or double density is in place.

- Software select single or double density.
- Phase-locked-loop and write precompensation for reliable data recovery and storage.
- On-board phantom bootstrap PROM is disabled after bootstrap operation so all 64K memory address space is available to user.
- DMA in single or double density permits multi-user operation.
- Extended addressing provides 8 extra address bits, permitting direct transfer anywhere in a 16 megabyte address range.
- Select up to 4 drives, single or double sided.
- New BIOS for CP/M included on single-density diskette.

Assembled and Tested$425

CP/M is a reg. trademark of Digital Research.



Tarbell Electronics 950 Dovlen Place, Suite B,Carson, Ca. 90746
(213) 538-4251 (213) 538-2254

CIRCLE INQUIRY NO. 76

From the Fountainhead

By Adam Osborne

I have started to receive feedback on computer software that has been developed under government grants and is now being sold. Government grants were supposed to fund the development of software that would subsequently be placed in the public domain.

It appears as though there is a bigger scandal in this one than meets the eye.

But before we brand everyone a crook because at some point they received government money, it is important to note that no problem is all black or all white, we are always dealing with some shade of gray.

STOIC is a computer language that was developed at the Massachusetts Institute of Technology (MIT). MIT will sell you STOIC for a small handling fee; and they request that users register with the Institute. So far as I'm concerned, this is a perfectly legitimate policy. Because a university (or any other institution) received government funding to develop a piece of software, that does not oblige them to distribute and support the software indefinitely, at no charge. The principle here is that you are getting at no charge what your taxes paid for, but you are paying for services that the government did not fund. I have no quarrel with MIT's request that users register; if the Institute is going to make any attempt at servicing a customer base, they had better know who the customer base is.

Unfortunately MIT provides STOIC on 9-track tape. As delivered, it is not easy to use on typical microcomputer systems. Jeffrey L. Zurkow (of Avocet Systems, 804 South State St., Dover, Delaware 19901) will sell you STOIC on floppy disk, ready to run on most 8080A based microcomputer systems. Is that OK? Sure — so long as customers know they are paying for installation convenience, not for the product itself.

Gary Kildall developed some portions of his CP/M operating system while he was at the Naval Post Graduate School in Monterey. Gary subsequently did a great deal of work at his own expense generating the product now sold as CP/M. The question here is whether the price charged for the whole product is justified by the additions made at the seller's expense. And in the case of CP/M there is no question that the price is worth the addition. I have a copy of a letter, dated June 26, 1975, and signed by Gary Kildall, stating that public domain modules of CP/M are: CCP, BDOS, PIP, LOAD, DUMP and IOLIB. Of course, 1975 versions of these modules are in the public domain — not enhancements made subsequently at Gary's expense. Anyone who disagrees with this position is free to take the public domain modules of CP/M, add their own enhancements, then try to compete with CP/M.

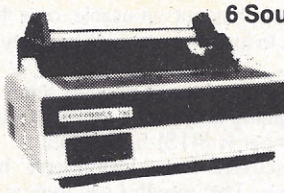
Gary Kildall is involved in one controversy surrounding an early version of CP/M that was sold to Lawrence Livermore Laboratories and supposedly went into the public domain with the sale. I am not prepared to comment on the validity of either side in this controversy; however, the version of CP/M involved is not current.

There are two types of operation that I believe we should pay careful attention to. Both involve some degree of deception.

First, there is the software package which is almost entirely in the public domain. Customers pay a lot of money for a very small addition to this public domain package, because they are not aware of the large part that is really free, as compared to the small part which is not. If customers know what they are paying for, and are willing to pay the price, that is fair. If customers pay the price because they have been kept in ignorance, that is not fair.

HARDSIDE™

6 South Street • Milford, New Hampshire 03055 • (603) 673-5144

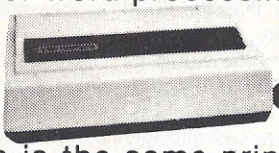


PRINTERS

CENTRONICS 753-2...

This high speed, high density, dot matrix printer (180 CPS) features an 18 x 9 dot matrix and proportional spacing. 132 characters per line. Ideal for word processing and all business uses. Includes connecting cable.

List Price $3295.00 **HardSide Price $2795.00**



CENTRONICS 730...

This is the same printer as the new Radio Shack "Line Printer II". Prints at 100 characters per second, 8 inch lines of 80 characters each. Features upper and lower case letters, with wide letters under software control. Operates as both a friction feed and a pin feed printer. Uses paper up to 9½ inches wide. Can also handle a single sheet of paper. Includes connection cable.

List Price $999.00 **HardSide Price $899.00**



QUICK PRINTER II...

A "mini" line printer priced to fit everyone's budget. Connects directly to the keyboard without an expansion interface. Software selectable for 16 or 32 character lines of both upper and lower case letters. Automatic "wrap-around"so no data is lost.

List Price $219.00 **HardSide Price $197.00**

CALL TOLL FREE 1-800-258-1790

HARDSIDE™

EXPANSION INTERFACE:	LIST PRICE	OUR PRICE
.... OK	$299	$269
.... 16K	$448	379/403
.... 32K	$597	479/537
DISK DRIVES:		
Percom, TFD-100, 40-track		$399
Percom, dual TFD-100		$795
Percom, TFD-200, 77-track		$675
2-drive cable		$29
4-drive cable		$39
Radio Shack, -0	$499	$469
Radio Shack, -1, 2, 3	$399	$459
ACCESSORIES:		
Telephone Interface	$199	$179
16K Memory Kit		$99
16K Memory Kit for E.I.		$95
RS232-C Interface	$99	$89
TRS-232 Interface		$49
Data Dubber		$49

HARDSIDE™

TRS-80 COMPUTERS:	LIST PRICE	OUR PRICE
Level-I 4K	$499	$449
Level-I 16K, w/keypad	$729	$659
Level-II 4K	$619	$559
Level-II 16K		$669
Level-II 16K, w/keypad	$849	$769
PRINTERS:		
Line Printer III	$1999	$1849
Centronics 779-2	$1598	$1095
Line Printer II	$999	
Centronics 730	$999	$899
Centronics P1 *	$499	$419
Quick Printer II	$219	$197
QP-II Exp. Int. Cable	$20	$18
* Cable required	$39	$35

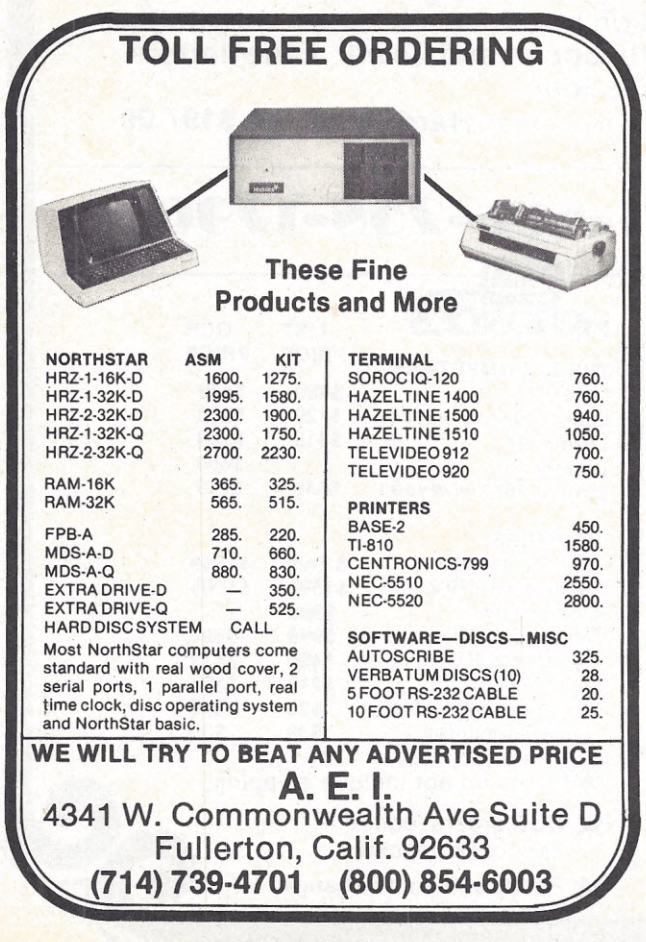
✶ Prices do **not** include shipping

✶ COD orders require 25% cash deposit

✶ Prices subject to change

*TRS-80 is a trademark of Radio Shack and Tandy Corp.





CIRCLE INQUIRY NO. 1

Then there is public domain software which is unobtainable in any form outside of commercial package. In particular, when the commercial package happens to be sold by a company that has close ties to university faculty members who developed most of the software under government grants, we need to start asking questions.

And what about the software that has been in development in universities for years, but is not made available to anyone, on the grounds that it is incomplete? We all know that no software package is ever complete. The question is this: is the software really being kept out of our hands because it is not yet usable, or is the software being kept out of our hands to further the private interests of the software developers?

I intend to explore these questions further in future columns. Those of you who have anything to say on the subject, I urge you to call me. My telephone number is (415) 548-2805.

In a recent column, I discussed Texas Instruments' history as a microprocessor manufacturer. This month I will look at Intel.

Although Intel first manufactured a four-bit microprocessor, the 4004, it was the 8008 which led to the 8080A, and gave us the microprocessor industry as we know it today. I therefore look upon the 8008, rather than the 4004, as the ancestor of today's microprocessor industry.

Intel developed the 8008 for Datapoint. Datapoint designed a very primitive CPU architecture, which they contracted with Texas Instruments and Intel to build as a single chip. Datapoint intended to use the chip in low cost intelligent computer terminals. Texas Instruments was unsuccessful in its attempt to build a CPU to Datapoint's specifications. So was Intel; Intel built a CPU that met all of Datapoint's requirements, except that it was a factor of ten too slow. Datapoint therefore built its CPU out of standard electronic components and sold the product as the Datapoint 2200. Intel was left with a slow, one-chip CPU which they called the 8008. Very few people realize that the Datapoint 2200 CPU has exactly the same assembly language instruction set as Intel's 8008, for the simple reason that the 8008 was originally designed to be the CPU for the 2200.

Intel started selling 8008's in 1973. They were so stunned with the success of this product that they quickly redesigned it, taking out the most obvious limitations; the result was the 8080. But the 8080 had immediate problems, in particular design errors in its interrupt logic, which were ironed out in the 8080A. The 8080A first entered the market in the Spring of 1974. At that time, Intel hired Bill Davidow to market the 8080A. This caused problems within Intel, since Ralph Ungermann and Federico Faggin had been in charge of designing the 8080A. They immediately ran into conflict with Bill Davidow over who would be in charge of what. The result was Zilog. Messers Ungermann and Faggin could not live with the separation of duties, so they left Intel and shortly thereafter formed Zilog. There was plenty of animosity surrounding these events. This animosity is not surprising, since so many of the geniuses that have made the semiconductor industry are prima donnas. This trait manifests itself in the strangest ways. For example one of the most bitter points of contention surrounding the 8080A involved its assembly language mnemonics, which Bill Davidow's people completely revised from those previously established by Ungermann and Faggin. Ungermann and Faggin's mnemonics subsequently surfaced within the Z80 assembly language. In particular the extensive use of the LD mnemonic to cover a vast array of instructions that move data, was a feature that Ungermann and Faggin battled for, but Davidow's crew threw out.

We can, in hindsight, argue as to whether Intel went on to become the leading microprocessor manufacturer based on luck or a good understanding of the market, but what is clear is that they did become the leading manufacturer. Rather than speculate on how much this was luck and how much was good judgement, it might be more productive to look at what Intel did to get them their current enviable position. In three words: they provided support. No one else did. Most semiconductor manufacturers looked upon a microprocessor as another electronics component, supported by a clerk who knew how to read numbers and prices out of a catalog. Intel, under Davidow, was the first semiconductor manufacturer to start building development systems. Intel was the first manufacturer to invest in a large software group. Intel was the first manufacturer to spend money on documentation and support, treating a microprocessor as a minicomputer, not a semiconductor device. This, more than anything else, propelled Intel to its number one position. If you look at the 8080A, the 8085, the 8086, or any of the support parts

surrounding these microprocessors, you will find they are all very adequate — but so are products manufactured by Intel competitors. In fact, many competitors' products were quite superior. I will describe some of these competitors, and why they failed, in future columns.

But Intel learned well the lesson that IBM has taught us all. To be number one your product need not be technically the best, it need not even be technically good; it must be adequate for the jobs into which it is being sold, it must be well supported and readily available. That is what counts, and that is what Intel has done better than anyone else.

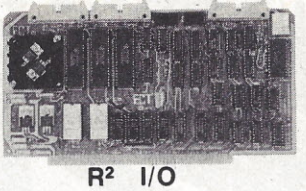
Many members of the MITA Board of Directors, and other influential people in the microcomputer industry, appear enamored with "The Source" — a communications and database system available to all microcomputer users. The idea sounds like a good one. Any microcomputer user can acquire a modem and turn his or her microcomputer system into a terminal capable of accessing a diverse variety of data files. You can also communicate with other Source users, transmitting messages and receiving replies via personal letter boxes.

I have encountered two problems with "The Source."

First of all, the data files made available to Source subscribers are fairly standard stuff, available to almost anyone who knows how to go about accessing data files. I could not identify any data file available strictly to Source subscribers. But a more important criticism I have with The Source is that their personnel are not very helpful. I have not subscribed to The Source because of unanswered telephone calls and generally poor communications with Source personnel. If this occurred when I was seeking information as a potential subscriber, what will it be like after I have paid my dues and am on the hook? Only when I identified who I was, did I get any response — and then they were all over me. But I would be interested in knowing what the potential for such a communications system might be. For example, would stores like to use The Source (or an equivalent database system) as a means of communicating with each other, with customers and with vendors? If store owners have any comments, please send them to me.□

Contact Dr. Osborne at P.O. Box 1234, Cerritos, CA 90701.

BUILDING BLOCKS FOR MICROCOMPUTER SYSTEMS, CONTROL & TEST EQUIPMENT



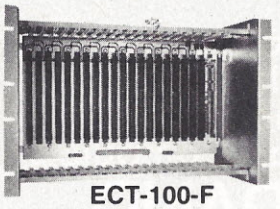
R² I/O

2K ROM 2 K RAM
3 SERIAL PORTS 1 PARALLEL PORT



TT-10

TABLE TOP MAINFRAMES



ECT-100-F

RACKMOUNT CARD CAGES

POWER SUPPLIES, CPU's, MEMORY, OEM VARIATIONS

SPECIALIZING IN QUALITY MICRO-COMPUTER HARDWARE

INDUSTRIAL
EDUCATIONAL
SMALL BUSINESS
PERSONAL

ELECTRONIC CONTROL TECHNOLOGY

763 RAMSEY AVE.
HILLSIDE, N.J. 07205

(201) 686-8080

CIRCLE INQUIRY NO. 25

EXTEND YOUR WORD PROCESSING POWER

ELECTRIC PENCIL* + TEXTWRITER

CREATE AND EDIT DOCUMENTS WITH ELECTRIC PENCIL
FORMAT AND PRINT THEM WITH TEXTWRITER

- CONTRACTS & SPECIFICATIONS
 Standard paragraphs or sections stored in files and inserted by name when printed
- PERSONALIZED FORMLETTERS
 Names, addresses, etc. replaced by entries from a mail list file or from the keyboard
- BOOKS & ARTICLES
 Footnotes collected & printed at page bottom, chapters kept in separate files chained together when printed
- REPORTS & MANUALS
 Table of contents & alphabetized index printed automatically

Formats files created by your editor • Works with any terminal & printer • Written in assembly language for high speed
Available on all common 8" & 5¼" disks • Versions for CP/M, TRS-80 CP/M, Northstar DOS and Micropolis MDOS • Compatible with: Electric Pencil I & II, Wordmaster*, Wordstar*, ED, Edit-80, PDS Edit, Micropolis Lineedit, and many more.

Ask Your Dealer For A Demonstration

Textwriter III $125
Manual alone - $15
Add $2 per order for shipping
Foreign airmail $7.50
$1 extra UPS COD
California residents add 6½% sales tax

*Wordmaster and Wordstar are trademarks of MicroPro, Int.; Electric Pencil is a trademark of Michael Shrayer Software; CP/M is a trademark of Digital Research.

For Further Information Call
(415) 455-4034



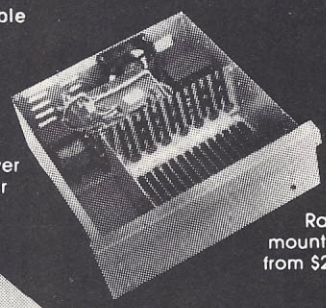
1492 Windsor Way, Livermore, CA 94550

CIRCLE INQUIRY NO. 57

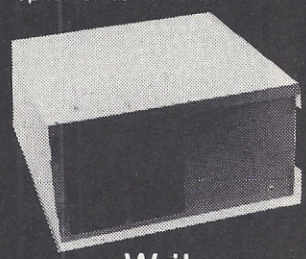
Main/Frames from $200

Main/Frames from $200

- 14 Basic Models Available
- Assembled & Tested
- Power Supply: 8v@15A, ± 16v@3A
- 15 Slot Motherboard (connectors optional)
- Card cage & guides
- Fan, line cord, fuse, power & reset switches, EMI filter
- 8v@30A, ± 16v@10A option on some models



Rack mounted from $200



8" Floppy Main/Frame (includes power for drives and mainframe) from $365

Write or call for our brochure which includes our application note: 'Building Cheap Computers'

INTEGRAND

8474 Ave. 296 • Visalia, CA 93277 • (209) 733-9288

We accept BankAmericard/Visa and MasterCharge

CIRCLE INQUIRY NO. 38

$159.95

OSBORNE BUSINESS SOFTWARE

General Ledger	- $59.95
Accounts Payable & Accounts Receivable	- " "
Payroll	- " "
All Three - Complete	- $159.95

format - 8" single density CP/M diskette

requires CP/M & CBASIC - 2

manuals - $20. each

VANDATA

17541 stone avenue n. • seattle, washington 98133

(206) - 542 - 8370

CIRCLE INQUIRY NO. 77

JURISPRUDENT COMPUTERIST



By Leonard Tachner
Attorney-at-Law

United States Patent [19] [11] 3,921,161

Baer [45] **Nov. 18, 1975**

[54] **PREPROGRAMMED TELEVISION GAMING SYSTEM**

[75] Inventor: **Ralph H. Baer**, Manchester, N.H.

[73] Assignee: **Sanders Associates, Inc.**, Nashua, N.H.

[22] Filed: **Feb. 22, 1974**

[21] Appl. No.: **445,022**

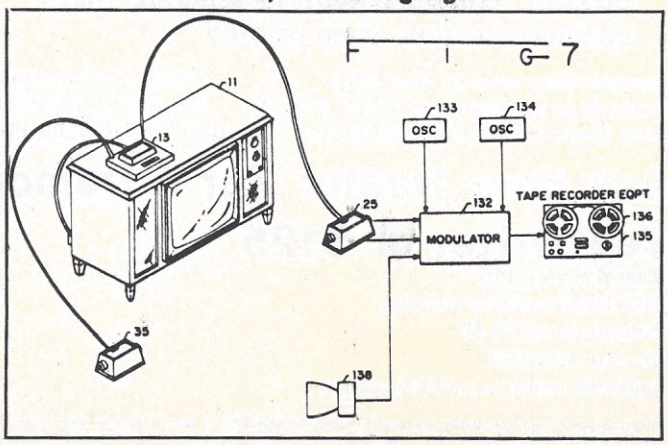
Related U.S. Application Data

[63] Continuation-in-part of Ser. No. 365,000, May 29, 1973, abandoned.

[57] **ABSTRACT**

Apparatus for playing games on the cathode ray tube screen of a television receiver is disclosed comprising a gaming device having components for providing location controllable spots on the screen, a source of prerecorded modulated audio frequency control signals, and means for demodulating those control signals and for supplying them to selected gaming device components to thereby provide gaming capabilities where the progress of a game is in part controlled by the recorded signals and in part controlled by the actions of a player. The signal source may be a magnetic tape or record/disc player and a selected prerecorded game tape or disc with control signals recorded thereon as a plurality of modulated audio frequency signals to thereby allow signal separation based on frequency and waveform control techniques and subsequent demodulation of each so separated carrier to provide a plurality of control signals.

25 Claims, 10 Drawing Figures



BACKGROUND OF THE INVENTION

The present invention relates generally to electronic game devices and more particularly to electronic game devices which generate signals for displaying spots on a television receiver screen. More specifically the present invention relates to such a televison gaming device where certain of the visible variables associated with one or more of the spots is under the control of a player while certain of the visible variables are automatically controlled by a selection preprogrammed game tape.

SUMMARY OF THE INVENTION

The foregoing as well as numerous other objects and advantages of the present invention are achieved by providing apparatus for playing games on the screen of a cathode ray tube including means for generating and displaying on the screen game spots having a plurality of visible variables such as horizontal and vertical position, size, intensity, and color associated therewith; player actuable means coupled to the generator for controlling certain ones of the variables; and preprogrammed means such as a magnetic tape and tape reader coupled to the generator for automatically controlling certain ones of the variables in accordance with information recorded thereon to thereby allow active participation by a player in determining the progress of the game while a part of the progress of the game is predetermined by the information recorded on a selected game tape. A "library" of such tapes for numerous games may be provided. The game progress determining information may be recorded on the game tape as frequency, amplitude or pulse width modulated audio frequency signals and the game tape may be played on a standard tape recorder requiring only the addition of some type of demodulator and the appropriate means for coupling the control signals to the prior art gaming device. Audio information or additional control signals may be recorded in a second track on the tape if desired and a plurality of control signals may be recorded on a single track by frequency multiplexing or other multiplexing scheme.

It is accordingly a specific object of the present invention to provide prerecorded and home recorded game spot control for one or more of the spots in a television gaming device requiring a minimal amount of equipment in addition to equipments already available.

BRIEF DESCRIPTION OF THE DRAWING

The aforementioned and other objects, features and advantages of the present invention will become more apparent from the following detailed description thereof when considered in conjunction with the drawings wherein:

FIG. 7 is a block diagram depicting the means for creating a recorded tape for use in the preprogrammed control unit;

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 7 shows in block diagram format a representative arrangement of equipment required to create the prerecorded tape or record. This arrangement can be used for single or multiple track recording, regardless of the type of modulation employed and irrespective of the presence or absence of audio (voice, music, etc.) accompaniment. One of the player control units 25 or 35 is used to generate the horizontal and vertical positioning signals. The horizontal and vertical outputs are applied to modulator 132, which has inputs from oscillators 133 and 134 to generate the two band signals. If audio, such as voice or music, is to be recorded, it is ap-

3G LIGHT PEN PET APPLE TRS-80 (Level II)



- **Bypass the keyboard** and interact directly with the screen. Makes your computer more versatile for the novice as well as the more sophisticated user.
- The use of the 3-G Light Pen is limited only by your imagination! Use it to experiment with **graphics**, display a menu for quick **data retrieval**, as a **teaching tool** with your child, to proportion **recipes** instantly, or to play unique **games**.
- At a major medical center an anesthesiologist uses our pen to select proper dosages. In Holland they use it to create graphics. A man in New York uses it to teach his pre-school age daughter how to match capital with lower case letters. Teachers are using it in science and foreign language classes.
- **NO ASSEMBLY NECESSARY. READY TO PLUG IN.** Detailed sample programs included. Complete documentation so you can write your own programs in **BASIC.** No machine language coding necessary. All Professional models plug into machine ports and don't require batteries. Economy model plugs into tape recorder and batteries are included.
- **YOU GET:** 1) 3-G Light Pen
 2) Demo-Game Cassette (with Professional TRS-80 and Apple).
 3) Sample Program.
 4) Complete documentation and instructions.

MANY GAMES AND OTHER LIGHT PEN SOFTWARE AVAILABLE.

———MAIL COUPON OR CALL TODAY FOR IMMEDIATE DELIVERY———

3G COMPANY, INC. DEPT. IA (503) 662-4492
RT. 3, BOX 28A, GASTON, OR 97119

[] TRS-80 Economy **$19.95** [] TRS-80 Professional **$34.95** [] PET Professional **$29.95** [] Apple Professional **$32.95**

Yes. I want to make my computer more versatile. Rush me _____ 3-G Light Pens. (Add $1.50 for mailing and handling - $6.00 foreign.)

Enclosed is: [] check or money order [] Master Charge [] Visa

Card No. _____ Exp. date _____

NAME _____

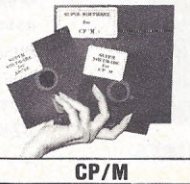
ADDRESS _____

CITY _____ STATE _____ ZIP _____

REMEMBER, 3G OFFERS A 30 DAY UNCONDITIONAL MONEY BACK GUARANTEE!

CIRCLE INQUIRY NO. 30

DISCOUNT SOFTWARE



It's about time. You can now afford quality software. We will beat any nationally advertised price. Watch for our ads!

CP/M

OSBORNE AND ASSO. business software in CBASIC2.

- General Ledger
- Payroll/Cost
- Acct. Rec./Pay

All 3 $98.00 $118.00 after 4/15/80.

Complete with manuals and source code.

WORDSTAR by MicroPro.....**$399.00**

PEACHTREE business software.

General Ledger • Payroll
Acct. Rec. • Acct. Pay.
Inventory

Save 35%, $650.00 ea.
Special: All 5 $2500.00

CP/M is a TM of Digital Research Corp.
Apple II is a TM of Apple Computers, Inc.
CBASIC2 is a TM of Software Systems
VisiCalc is a TM of Personal Software, Inc.
EasyWriter is a TM of Cap'n Software, Inc.

APPLE II

VisiCalc.....**122.00**

EasyWriter word processing software
80 column version with
SUPER 'R' TERM1 board **15% off.**

VISA • MASTERCHARGE
CALL **TOLL FREE** 7 DAYS
24 HOURS A DAY
1-800-854-2003, EX. 823
California only 1-800-522-1500 EX. 823
The Discount Software Group
1610 Argyle Ave., Bldg. 102
Los Angeles, CA 90028
(213) 461-3127

CP/M users: specify disk systems and formats. Most formats available.

Add $2.50 postage and handling per each item. California residents add 6% sales tax. Allow 2 weeks on checks. C.O.D. ok.

Prices subject to change without notice. All items subject to availability.

CIRCLE INQUIRY NO. 21

CP/M® SOFTWARE

Word Processing For CP/M

FMT Text Formatter *$75*

Use FMT and your text editor to convert your CP/M system to a powerful word processor. FMT features include automatic page headings and footings, page numbering, centering, underscoring, external file merging, and in-line console input. FMT works with any video, CRT, or hardcopy terminal and printer combination. Added capabilities for daisy-wheel printers: superscripting, subscripting, and half-line spacing.

Run Cromemco Software Under CP/M

ADAPT Software Interface *$50*

Now you can get Cromemco software to run on your CP/M system. ADAPT interfaces those powerful Cromemco packages (except for Multi-User BASIC) to any Z-80 based CP/M system without patching. ADAPT works without changes for any memory size.

Fast RATFOR

RATFOR (RATional FORtran) *$95*

RATFOR lets you write structured code that translates to Microsoft or Cromemco FORTRAN. TSW's RATFOR (RATional FORtran) precompiler runs at more than 1000 statements per minute. Documentation includes "Software Tools" book by Kernighan and Plauger. (ADAPT and RATFOR packages combined $125)

TSW

THE SOFTWARE WORKS
8369 Vickers
San Diego, CA 92111
(714) 569-1721

VISA and MasterCharge accepted.
® CP/M is a trademark of Digital Research.

CIRCLE INQUIRY NO. 69

plied, for example, from a microphone 138 to the modulator. The modulator 132 is of conventional design, either an amplitude modulator, frequency modulator or pulse width modulator, is well known in the art. The output from the modulator is applied to a recorder 135, recording on tape 136, or alternatively for recording on discs or the like.

By also coupling the output from the player control unit through the electronic circuitry 13, connected to the television receiver 11, the user may monitor the information being recorded on the tape by viewing the television receiver and seeing the positioning of the spot while he is making the tape.

Numerous other variations and modifications will suggest themselves to those of ordinary skill in the art and accordingly the scope of the present invention is to be measured only by that of the appended claims.

1. Apparatus for playing games on the screen of a cathode ray tube comprising:

means for generating on the screen a first spot having a plurality of visible variables associated therewith and representing a first moving game playing object;

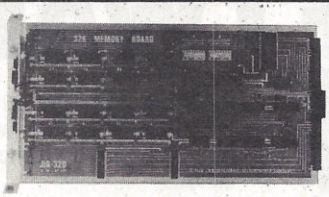
means for generating on the screen a second spot having a plurality of visible variables associated therewith and representing a second moving game playing object;

player actuable means for controlling at least one variable of at least one of the spots; and

prerecorded means having control signals recorded thereon in the form of a modulated audio frequency signal for automatically controlling at least one variable of at least one of the spots.

To contact the author write Jurisprudent Computerist, P.O. Box 1234, Cerritos, CA 90701.

HEATH® COMPATIBLE



DG-32D 32K RAM FEATURES:

- Plugs into Heath® H8 Computer.
- Ready to use. Fully assembled, tested & burned in.
- Operates with existing Heath® memory.
- Protected memory Output Buffers in the event of Address error.
- Utilizes popular 4116 RAM devices.
- Memory Address DIP switch changeable.
- Arranged as 4 Independent 8K Blocks.
- Low Power Consumption: Less than 6 watts, typical.
- Transparent Refresh.
- One year guarantee.
- Compatible with all current H8 peripherals.

D•G Electronic Developments Co. brings you a totally compatible, fully assembled and tested 32K RAM for Heath® H8 computers. The DG-32D has less than 6 watts power consumption. This allows you to add a full 32K bytes of Random Access Memory without taxing or replacing your computer's power supply. Engineered to plug-in and run without any user modifications, the DG-32D can be used with or without existing H8 RAM without modification. Protection of the memory output buffers is provided in the event of assigning two blocks to the same address space. The DG-32D is the ideal answer to expansion of the Heath® H8 computer. . . Low power consumption, low price, high capacity, total engineering and exacting production methods.

Heath® and H8 are registered trademarks of the Heath Corporation, Benton Harbor, Michigan.

BURNED IN 16K MEMORY CHIP SET
H88-H89 **$89.00**

ORDERING INFORMATION: DG-32D RAM available only from DG Electronic Developments Co., P.O. Box 1124, 1827 South Armstrong, Denison, Texas 75020. Check, money-order, VISA or Master Charge. Phone orders accepted on charge orders. NO COD's. Foreign orders add 30%. Texas residents add 5%. For VISA or Master Charge orders call 214-465-7805. $479.00 freight prepaid. Allow three weeks for personal checks to clear banks.

D·G ELECTRONIC DEVELOPMENTS CO.

32K RAM $479.00

AL BAKER'S GAME CORNER

By Al Baker

MY TRS-80 LOVES ME WHEN I YELL AT IT

The Robots are coming! This month we are going to play with one of the more interesting areas of Robotics research, voice input. A properly domesticated robot will need to understand and respond to some fraction of its owner's spoken language. Our make-believe robot is a TRS-80. Let's make it listen to us. Type in this program:

```
10 CLS
20 OUT 255,4
30 IF INP(255) > 127 THEN PRINT"1";
40 GOTO 20
```

Now follow these instructions carefully:

1. Connect the black plug of your cassette cable into your tape recorder's ear jack.
2. Do not connect the other plugs to your cassette recorder. Nothing should be plugged into your cassette recorder's AUX jack.
3. Open the lid on the cassette recorder by pressing the EJECT button.
4. Reach into the area under the lid of the cassette recorder and push in the button at the left rear and hold it in; now
5. Press down on the cassette recorder RECORD button. If you cannot, you are pushing the wrong button in step 4.
6. While holding down the RECORD button, press down the PLAY button.

Type RUN on the TRS-80 and start yelling. Amazing! The computer hears you! If you have kids and really want to have fun, bring them in. They'll love a chance to sound off to the computer.

ATTACK COMMAND ALPHA

This month's TRS-80 game is an exciting demonstration of talking to your computer. You control the gun at the bottom of the screen. Planes are flying overhead. Now yell "Fire." The gun fires. Yell "Wham." The gun fires. Yell anything. The gun fires.

Lines 90 and 100 create the two airplanes. Using the CHR$ junction to draw pictures was described in the article "Using TRS-80 Graphics Codes" by Woody Pope in the December issue of INTERFACE AGE. Lines 630 and 640 listen for your yell. You should recognize them from the short program at the beginning of this column. The remainder of the game is straightforward.

MORE ON TALKING

Now you know how to make your TRS-80 hear you. That's about it in BASIC. Clever programmers will think up more ways of using this feature, but recognizing speech is not one of them. But if you know TRS-80 machine language, this may not be true.

With a machine language program you should be able to sample a person's voice often enough to develop a binary list which represents a word. If the 7 bit in port 255 is 1, put a 1 in the list, otherwise put a 0. With some *very* clever programming and mathematical analysis beyond the scope of this column you could have the computer user type in a word and then say it. After several repeats, the computer would print the typed word almost every time it heard it.

C-10 **SHORT CASSETTES** 50 FT.



Qty.	Price
1	$1.00
10	$0.75
50	$0.65

Premium tape and cassettes acclaimed by thousands of repeat order microcomputer users. Price includes labels, cassette box and shipping in U.S.A. VISA and M/C orders accepted. California residents add sales tax. Phone (415) 968-1604.

MICROSETTE CO.
475 Ellis Street
Mt. View, CA 94043

CIRCLE INQUIRY NO. 50

DATA On-Site Data Acquisition To Analysis System

Introducing the low-cost DATA system which does on-site data acquisition and analysis. This new system can be adapted for your data taking needs, whether a solar experiment or equipment for a home-owners lab. The DATA system includes

HARDWARE
- Z-80 CPU / 32K RAM / 180K mass storage
- A/D - 64 channels (8 bit) software programmable gains (1-1000)
- Hardware automatic restart
- Real time clock

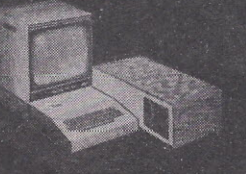
PERIPHERAL
- ACT IA - full ASCII
- Monitor - 64x16 format of 5x9 dot matrix

SENSORS
- 2 Lambda Pyranometers
- 5 temperature thermistor sensors
- 1 on/off relay

SOFTWARE
- Data collection computer program
- North Star DOS/BASIC

SYSTEM PRICE
$4000

Call or write
LONDE • PARKER MICHELS, INC.
Suite 202 • 7438 Forsyth,
St. Louis, MO. 63105
314/725-5501



CIRCLE INQUIRY NO. 43

By keeping several binary lists in memory at the same time, you could understand simple commands.

Who knows, you may be well on the way to building your own domestic Robot!□

Al Baker of The Image Producers can be contacted at P.O. Box 1234, Cerritos, CA 90701.

PROGRAM LISTING

```
10 '   ... ATTACK COMMAND ALPHA ...
20 '
30 '
40 '
50 ' A$=RIGHT MOVING AIRPLANE
60 ' B$=LEFT MOVING AIRPLANE
70 ' S ARRAY CONTAINS SCORES
80 '
90 A$=" "+CHR$(128+9)+CHR$(128+12)+CHR$(128+47)+CHR$(128+28)
100 B$=" "+CHR$(128+44)+CHR$(128+31)+CHR$(128+12)+CHR$(128+14)+" "
110 S(1)=0:S(2)=0
120 '
130 ' LOOP FOR EACH PLAYER, CLEAR SCREEN, AND SUPPLY 5 BULLETS
140 '
150 FOR I=1 TO 2
160 CLS
170 B=5
180 '
190 ' COME HERE AFTER EACH HIT AND DISPLAY GUN AND SCORES
200 '
210 SH=0
220 PRINT@991,CHR$(128+58);CHR$(128+16);
230 PRINT@512,"PLAYER";I;TAB(50)"SCORE";S(I);
240 PRINT@576,"PLAYER";3-I;TAB(50)"SCORE";S(3-I);
250 PRINT@960,"BULLETS";B;
260 '
270 ' MOVE PLANE RIGHT
280 '
290 FOR J=0 TO 59
300 PRINT@J,A$;
310 'IF NO BULLET SEE IF COMMAND TO FIRE, ELSE GO MOVE BULLET
320 IF SH=0 THEN GOSUB 630     ELSE GOSUB800
330 'IF HIT OR NO BUULETS, END LOOP
340 IF SH=2 THEN J=59
350 NEXTJ
360 'CLEAR PLANE FROM SCREEN
370 PRINT@0,CHR$(30);
380 'IF HIT OR NO BULLETS, BYPASS PLANE MOVING LEFT
390 IF SH=2 THEN 560
400 '
410 ' PLANE MOVING LEFT LOOP
420 '
430 FOR J=58 TO 0 STEP -1
440 PRINT@J,B$;
```


MICAH

OSBORNE BUSINESS SOFTWARE

in CBASIC2 *or* CROMEMCO 16K BASIC

★ *features* ★

- Four Complete Packages---
 - General Ledger
 - Accounts Receivable
 - Accounts Payable
 - Payroll with Cost Accounting
- Strong support from Osborne Manuals
- CBASIC2 runs under CP/M or under CDOS version 1.07 on Cromemco computers
- 16K BASIC runs on Cromemco computers
- Cursor addressing routines for Hazeltine, Lear Siegler and Cromemco (Beehive) Terminals
- Source Codes and Installation Instructions provided along with disks
- Automatic Command Start-up
- Easy to apply to all of your business and systems needs

★ *hardware required* ★

- One or more 8" or 5" Floppy Drives
- CRT with cursor addressing
- 132-Column Printer

$95 *per package*

TO ORDER
Add $5 for Shipping
Calif. add 6-1/2% Sales Tax
CREDIT CARDS ACCEPTED

• DEALER INQUIRIES INVITED •

MICAH'S PRODUCTS OF DISTINCTION

- OSBORNE BUSINESS SOFTWARE (in CBASIC2 & 16K)
- CBIOS (CP/M for Cromemco Computers)
- MICROPLOT (Versatile Printer Graphics)
- DÜP/1 (Disk Utilities for CP/M and CDOS)
- DÜP/2 (Disk Utilities for Cromemco)
- DÜP/3 (DÜP/2 Expanded)
- SPOOL (Cromemco Format Utility)
- EXPAND (Run Cromemco Software on CP/M)
- BLACKJACK (Tutorial Casino Action)
- DRIVE (Customized Printer Drivers)
- DBMS Conversion to NAD

• Call or Write for Free Catalogue and More Information •

★ We will Customize any of our programs at our Standard Consulting Rates ★

Ah! MICAH Satisfyin' Software

That turns your system on!

MICro Applications and Hardware

• CONSULTANTS and SOFTWARE DEVELOPERS •

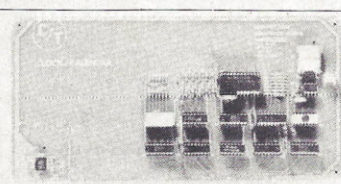
MICAH Box 22212 San Francisco, California 94122 USA phone: 415/664-0778

CIRCLE INQUIRY NO. 45

C/T

CT 102-A CLOCK/CALENDAR

S100 Bus



FEATURES

- Time of day in hours, minutes & seconds
- 24 hour time format
- Month & day date function
- Crystal controlled time base
- Fast & slow set functions ... allow rapid setting of time & date
- Output is latched BCD
- Simple read instructions allow simple interface to BASIC, CPM, etc.
- Will run with 4 MHz processors
- Optional battery backup capability
- Low battery drain in back-up mode 5-15 MA depending on voltage
- Can be located at any group of 4 I/O port addressed
- Basic E software example provided
- S100 bus signals used PWR, SOUT, SINP
- Easy interface to any S100 bus processor

The T102-A time date board can be used for any application requiring Time & date. A simple to use reading method allows the simplest BASIC language to be used
Outputs are latched BCD, just select the digit then read it.
Fast set & slow set functions provide for fast time & date setting.

Optional battery backup capability. A simple to construct charging circuit is included in the manual. Any 8 to 18V unregulated DC source is all that is required for backup.
The T102 comes assembled and tested. Each board includes an easy to use manual with basic software programs for setting and reading time and date.
Price **$149.95**

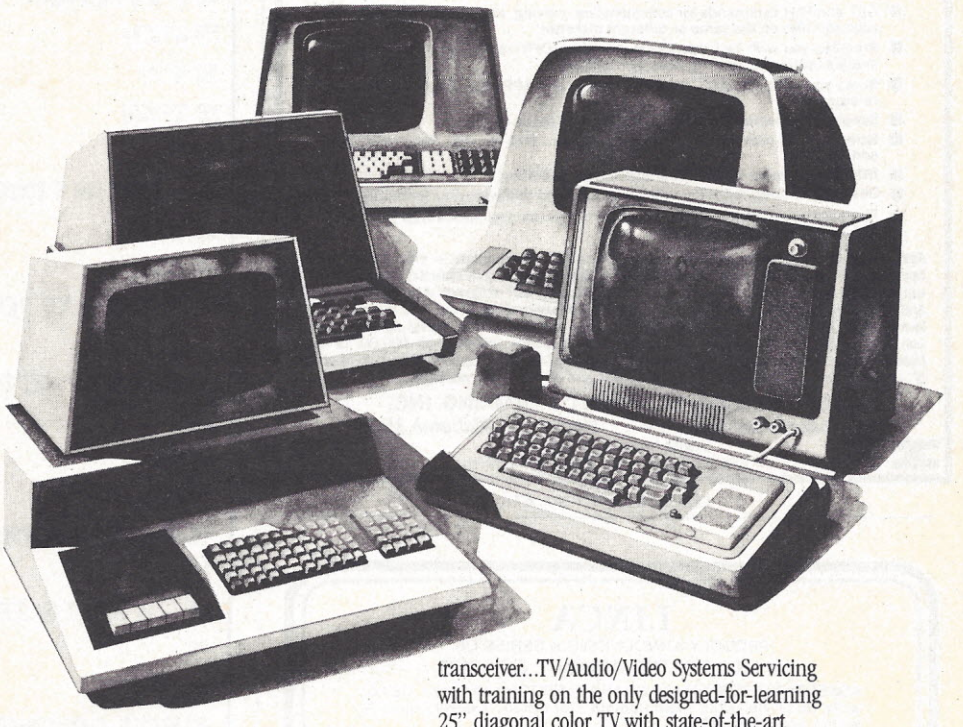
COMPU/TIME P.O. BOX 5343
HUNTINGTON BEACH, CA. 92646
(714) 536-5000

CIRCLE INQUIRY NO. 13

Everybody's making money selling microcomputers. Somebody's going to make money servicing them.

New NRI Home Study Course Shows You How to Make Money Servicing, Repairing, and Programming Personal and Small Business Computers

Seems like every time you turn around, somebody comes along with a new computer for home or business use. And they're being gobbled up to handle things like payrolls, billing, inventory, and other jobs for businesses of every size...to perform household functions like budgeting, environmental systems control, indexing recipes, and more.



Growing Demand for Computer Technicians... Learn in Your Spare Time

Even before the microprocessor burst upon the scene, the U.S. Department of Labor forecast over a 100% increase in job openings for the decade through 1985. Most of them *new* jobs created by the expanding world of the computer. NRI can train you at home to service both microcomputers and their big brothers. Train you at your convenience, with clearly written "bite-size" lessons that you do evenings or weekends without quitting your present job.

Assemble Your Own Microcomputer

NRI training includes practical experience. You start with meaningful experiments building and studying circuits on the NRI Discovery Lab.® Then you build your own test instruments like a transistorized volt-ohm meter, CMOS digital frequency counter...equipment you learn on, use later in your work.

And you build your own microcomputer, the only one designed for learning. It looks and operates like the finest of its kind, actually does more than many commercial units. But NRI engineers have designed components and planned assembly so it demonstrates important principles, gives you working experience in detecting and correcting problems. It's the kind of "hands-on" training you need to repair and service units now on the market.

Mail Coupon for Free Catalog No Salesman Will Call

Send today for our 100-page, full-color catalog. It describes NRI's new Microcomputer Technology course in detail, shows all equipment, kits, and lesson plans. And it also tells about other NRI courses... Complete Communications with 2-meter transceiver...TV/Audio/Video Systems Servicing with training on the only designed-for-learning 25" diagonal color TV with state-of-the-art computer programming. With more than a million students since 1914, NRI knows how to give you the most in home training for new opportunity. If coupon has been removed, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.



RUSH FOR FREE CATALOG

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
Please check for one free catalog only.
NO SALESMAN WILL CALL



All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Amateur Radio • Basic and Advanced
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating Including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council

171-040

CP/M® SOFTWARE TOOLS

NEW ED-80 TEXT EDITOR

ED-80 offers a refreshing new approach for the creation and editing of program and data files conversationally—and it saves you money. Its powerful editing capabilities will satisfy the most demanding professional—yet it can still be used by the inexperienced beginner.

Look at These Outstanding Features:

- FULL SCREEN window displays with forward and backward scrolling for editing your data a page-at-a-time, rather than line-by-line.
- Provides you with all the features found on the large mainframe and minicomputer editors, such as IBM, UNIVAC, CDC, and DEC.
- Commands include forward or backward LOCATE, CHANGE, and FIND; and INSERT, DELETE, REPLACE, APPEND, SAVE, PRINT, WINDOW, MACRO, TABSET, SCALE, DUMP, and others.
- Compatible with existing CP/M edit and text formatted files, with CBASIC, and with Microsoft's MBASIC, FORTRAN, COBOL, and ASSEMBLER.
- CHANGE commands allow you to make conditional changes and to use variable length strings.
- Designed for CP/M and derivative operating systems, including LIFEBOAT, CDOS, IMDOS, DOS-A, ADOS, etc.
- GET and PUT commands for concatenating, moving, duplicating, and merging your edit files on the same or different diskettes.
- Provides you with fast memory-to-memory COPY commands, and an intermediate buffer for copying lines over-and-over.
- Saves your last LOCATE, CHANGE, FIND, and APPEND command for easy re-execution.
- Simple line-oriented commands for character string editing.
- Safeguards to prevent catastrophic user errors that result in the loss of your edit file.
- INLINE command for your character-oriented editing.
- Designed for today's CRT's, video monitors, and teletypewriter terminals.
- Thoroughly field tested and documented with a comprehensive User's Manual and self-instructional tutorial.

And remember — in today's interactive programming environment — your most important software tool is your text editor. ED-80 is already working in industry, government, universities, and in personal computing to significantly cut program development time and to reduce high labor costs. Why not let ED-80 begin solving your text editing problems today? ED-80 is protected by copyright and furnished under a paid-up license for use on a single computer system. Single Density Diskette and Manual: $99.00, or the Manual alone: $20.00 (credited with purchase of the Diskette). Specify Disk make/model, 5" or 8", hard or soft sectored. ORDER NOW and we'll pay the postage!

SOFTWARE DEVELOPMENT & TRAINING, INC.
Post Office Box 4511 — Huntsville, Alabama 35802
Dealer Inquiries Welcomed
® CP/M is a trademark of Digital Research

CIRCLE INQUIRY NO. 67

LINCA

PROUDLY ANNOUNCING A SERIES OF
S - 100 STATIC MEMORY BOARDS

ASSEMBLED, TESTED and GUARANTEED* for one year. Call or write us or visit your local dealers* for detail. Delivery usually within one week ARO.

- **EXTENDED 32K STATIC RAM (ESRA32)**
 ESRA32-25 (250 ns): $829.00; -25A: $699.00
 ESRA32-45 (450 ns): $779.00; -45A: $659.00
- **EXTENDED 16K STATIC RAM (ESRA16)**
 ESRA16-25 (250 ns): $499.00; -25A: $399.00
 ESRA16-45 (450 ns): $469.00; -45A: $379.00
- **16K STATIC RAM (SRA16)**
 SRA16-25 (250 ns): $399.00; -25A: $349.00
 SRA16-45 (450 ns): $369.00; -45A: $329.00
- **8K STATIC RAM (SRA8)**
 SRA8-25 (250 ns): $219.00; -25A: $179.00
 SRA8-45 (250 ns): $199.00; -45A: $165.00

MAIN FEATURES:

1. ESRA32 & ESRA16 are extended with memory-management and bank-select and provided with POWER-ON and RESET control option.
2. External STAND-BY-POWER line provided without using any bus pin.
3. No jumper to install, no trace to cut by users for any possible configuration; just change the already installed jumper positions.
4. All address, data-in and data-out lines are fully buffered.
5. PHANTOM line option provided.
6. Can be deactivated in any combination of 4K pages with hardware jumpers except SRA8 model.
7. All address, data & control lines are Schmitt-triggered except SRA8 model.
8. TMS40L44 for ESRA32, TMS4044 for ESRA16 & SRA16 & 2102AL for SRA8.
9. Addressing at 4K boundaries except SRA8 at 1K boundaries.
10. Hardware WRITE-PROTECT in any combination of 4K pages for ESRA16 & SRA16.
11. Operable with or without front panel.
12. 0-1-2 wait cycles jumper selectable for ESRA16.

*90 days for suffix A versions that are not available from dealers. VISA, MASTER-CHARGE or COD acceptable. California residents add 6% tax. Delivery free within USA. Return postpaid without damage within 2 weeks if not satisfied.

Prices subject to change without notice

LINCA INDUSTRIES, INC.
22119-½ S. Vermont Ave., Torrance, California 90502
Phone: (213) 533-8258.

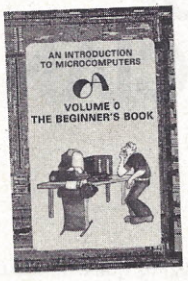
CIRCLE INQUIRY NO. 41


```
450 IF SH=0 THEN GOSUB630     ELSE GOSUB 800
460 IF SH=2 THEN J=0
470 NEXTJ
480 PRINT@0,CHR$(30);
490 IF SH=2 THEN 560
500 GOTO250
510 '
520 ' WE EITHER HAD A HIT OR ARE OUT OF BULLETS
530 ' IF NO BULLETS DO PLAYER LOOP
540 ' IF HIT GO UP FOR ANOTHER AIRPLANE
550 '
560 IF B=0 THEN 580
570 GOTO210
580 NEXTI
590 GOTO150
600 '
610 ' LISTEN FOR A COMMAND TO FIRE
620 '
630 OUT255,4
640 IF INP(255)<128 THEN RETURN
650 '
660 ' I HEAR YOU.  SET FIRE FLAG, DECREMENT BULLETS AND PRINT
670 '
680 SH=1
690 B=B-1
700 PRINT@960,"BULLETS";B;
710 '
720 ' PUT BULLET ON SCREEN
730 '
740 L=927
750 PRINT@L,CHR$(128+40);
760 RETURN
770 '
780 ' MOVE BULLET
790 '
800 PRINT@L," ";
810 L=L-64
820 ' IF STILL LOWER THAN AIRPLANE, DONE
830 IF L<64 THEN 910
840 PRINT@L,CHR$(128+40);
850 RETURN
860 '
870 ' BULLET IS AT PLANE'S ALTITUDE
880 ' IF HIT, UPDATE SCORE
890 ' IF HIT OR NO BULLETS, SET SHOOT FLAG TO 2
900 '
910 SH=0
920 IF (L>J)AND(L<J+5) THEN S(I)=S(I)+1 : SH=2
930 IF B=0 THEN SH=2
940 RETURN
```

MICROCOMPUTER BOOKSHELF

AN INTRODUCTION TO MICROCOMPUTERS

Volume 0 — The Beginner's Book
by Adam Osborne



This book introduces computer logic and terminology to the complete beginner in the field of microcomputers. Numerous illustrations and photographs combine with clear, easy-to-follow text to provide an elementary but broad-based background.

#26-8 $7.95

Volume 1 — Basic Concepts
by Adam Osborne

A must for anyone in the computer field, this best-selling text explains hardware and programming concepts common to all microprocessors.

#02-0 $9.50

Volume 2 — Some Real Microprocessors
by Adam Osborne et al.

This unique reference provides objective descriptions of virtually every microprocessor on the market today. Lets you know what's available, how they work (or don't work), and how to use them. Loose-leaf. Binders and yearly updates (six issues) sold separately.

Vol. 2 book, 1978 ed.	#15-2	$25.00
Vol. 2 binder	#16-0	$ 5.00
Vol. 2 1978/79 updates	#97	$25.00
Vol. 2 1979/80 updates	#94	$25.00

Volume 3 — Some Real Support Devices
by Jerry Kane et al.

Same objective, in-depth coverage as Volume 2, but applied to support devices: memory, data converters, data communication devices, direct memory access controllers, busses, and much more. Loose-leaf. Binders and yearly updates (six issues) sold separately.

Vol. 3 book, 1978 ed.	#18-7	$15.00
Vol. 3 binder	#19-5	$ 5.00
Vol. 3 1978/79 updates	#98	$25.00
Vol. 3 1979/80 updates	#95	$25.00

RUNNING WILD - THE NEXT INDUSTRIAL REVOLUTION

by Adam Osborne

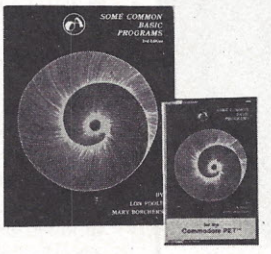
A 24-hour work week? Bionic man? Job redistribution? No one should miss this enlightening survey of the microelectronics industry and forecast for the future. Photos and illustrations. Intended for all audiences.

#28-4 $3.95

OSBORNE SOFTWARE

Some Common BASIC Programs
by Lon Poole et al.

76 short practical programs, most of which can be used on any microcomputer with any version of BASIC. Complete with listings, remarks, descriptions and examples. Special PET cassette and TRS-80 cassette versions.



SCBP book	#06-3	$12.50
SCBP PET cassette	#25-X	$15.00
SCBP TRS-80 cassette	#32-2	$15.00

NEW

CBASIC BUSINESS

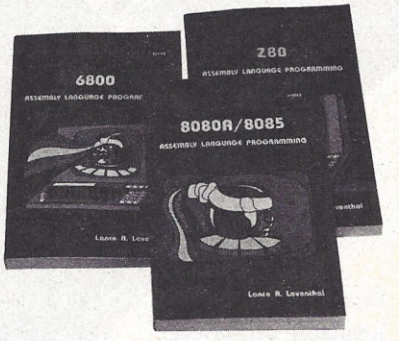
Payroll with Cost Accounting - CBASIC	#22-5	$20.00
Accounts Payable/Receivable - CBASIC	#23-3	$20.00
General Ledger - CBASIC	#24-1	$20.00

ASSEMBLY LANGUAGE PROGRAMMING

8080A/8085 • 6800 • Z80 • 6502

COMING 6809 • Z8000

These books explain assembly language programming, the functions of assemblers and assembly instructions, and basic software development concepts. Numerous practical programming examples are included for each. All books by Lance Leventhal.



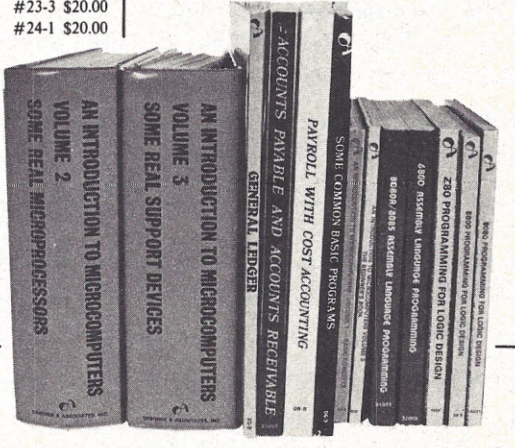
6502 ALP	#27-6	$12.50
Z80 ALP	#21-7	$12.50
6800 ALP	#12-8	$12.50
8080A/8085 ALP	#10-1	$12.50

6809 ALP (check box below to be notified of availability)
Z8000 ALP (check box below to be notified of availability)

Also:

Z80 Programming for Logic Design	#11-X	$9.50
6800 Programming for Logic Design	#05-5	$9.50
8080 Programming for Logic Design	#04-7	$9.50

OSBORNE/McGraw-Hill



Book/Cassette #	Price	Quantity	Amount
#			
#			
#			
#			

Calif. residents add 6% sales tax.
S.F. BART residents add 6-1/2 % sales tax.
No tax on update subscriptions.

Subtotal
California residents tax
Shipping
TOTAL

SHIPPING (Shipping for large orders to be arranged)
- ☐ All foreign orders $4.00 per item for airmail
- ■ No charge in the U.S. on update subscriptions.
- ☐ $0.75 per item 4th class in the U.S. (allow 3-4 weeks)
- ☐ $1.25 per item UPS in the U.S. (allow 10 days)
- ☐ $2.50 per item special rush shipment in the U.S.

For faster shipment or credit card, phone (415) 548-2805

☐ Please notify me when 6809 ALP is available.
☐ Please notify me when Z8000 ALP is available.

NAME:_____

ADDRESS:_____

CITY:_____ STATE:_____

ZIP:_____ PHONE:_____

OSBORNE/McGraw-Hill
630 Bancroft Way, Dept. C4
Berkeley, CA 94710
(415) 548-2805
TWX 910-366-7277

S1025

The BASIC Language Source Book

Four Complete BASICs for Microcomputers



First time available anywhere: Source Code Listing of Uiterwyk 4K BASIC for the 6800

Published in its entirety Lawrence Livermore BASIC for 8080, considered to be the most important microcomputer BASIC.

PLUS: Two additional powerful Tiny BASICs. Dr. Wang's Palo Alto Tiny BASIC and NIBL: National Industrial BASIC Language — Tiny BASIC.

Additional Bonus: Complete Index of Software published in INTERFACE AGE.

Yes, please send me _____ copies of the Best of INTERFACE AGE Volume 1 @ $12.95 each (Total $_____) to:

Name (print) _____

Address _____

City _____ **State**_____ **Zip**_____

Please add $.75 for shipping and handling charges.

Mail to:
DATA DYNAMICS TECHNOLOGY
P.O. Box 1217, Cerritos, CA 90701

☐ VISA ☐ master charge ☐ AMERICAN EXPRESS Cards Welcome

_____ ☐ **Check or M.O.**

Exp. Date_____ **Signature** _____

For European delivery contact **INTERFACE AGE EUROPE**
Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany

THE MICRO-MATHEMATICIAN

By Dr. Alfred Adler

FOURIER ANALYSIS — Part 3

In Part 1 we addressed the problem of analyzing quasi-periodic functions and tentatively selected Fourier series as the most promising tool. We learned that an arbitrary periodic function can be represented as accurately as desired by a Fourier series, and that the Fourier coefficients can be easily determined by use of the Euler formulas. Conversely, knowing the values of the Fourier coefficients, it is possible to reproduce or synthesize the original function, and that, in general, the more of the coefficients we have evaluated, the more accurately we can reproduce the original function. Various types of symmetry were discussed and a number of examples of simple periodic wave forms were examined.

Last month we presented Program 4YESERES, and used it to analyze and synthesize the simple waves discussed in Part 1. We studied the effects of varying the wave sampling rate on the accuracy of the Fourier coefficients. At the same time we studied the effects of sampling rate, the phase angle, the assumed period, and the number of terms in the Fourier series on the accuracy of the synthesis.

MODIFICATIONS TO PROGRAM 4YESERES

Since the listing of Program 4YESERES was published last month, it has been found convenient to add a third column to the table of Fourier coefficients. This column is labeled C(N), where

$$C(N) = \sqrt{A(N)^2 + B(N)^2}$$

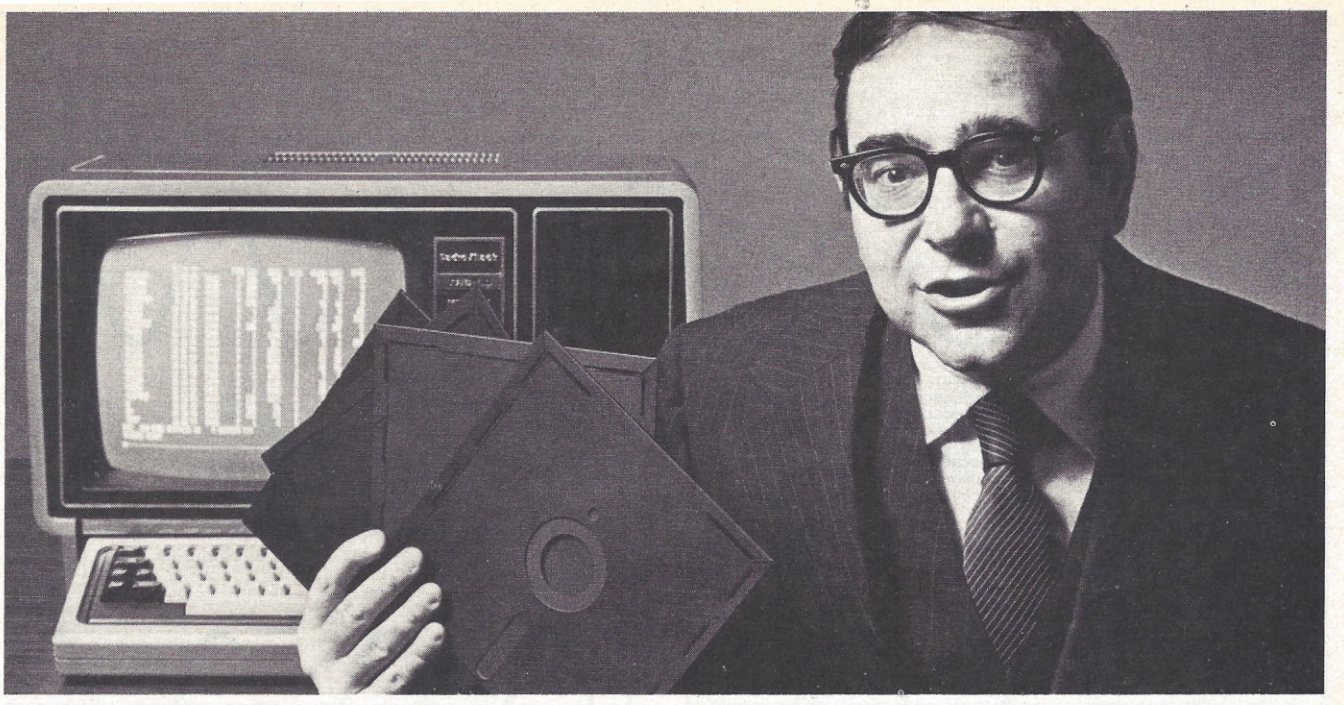
Recalling that A(N) is the magnitude of the cosine component of the Nth harmonic and that B(N) is the magnitude of the sine component of the Nth harmonic, it follows that C(N) is the magnitude of the Nth harmonic, without regard to phase. This is equivalent to saying that C(N) is the magnitude of the Nth coefficient in the complex Fourier series. The latter is a Fourier series expressed in terms of complex exponentials rather than trigonometric terms. The reader is referred back to this column in the May, 1979, issue (40-41), where complex numbers were discussed.

SINUSOIDS — ARBITRARY PERIOD

Last month we considered changes in period, but were very careful to make sure that all new periods remained multiples of 60 seconds. In the real world it is often difficult to determine just what the period is. Surely the results of the synthesis depend on the magnitude of the assumed period. Or do they? This question can be answered by running Program 4YESERES, choosing a sinusoid, and varying the desired period.

Figures 1 to 5 show such runs. In each case the curve is a sinusoid with a true period of 60 seconds and zero phase shift. The assumed periods, however, are 180, 195, 210, 225, and 240 seconds respectively. Note from the figures that even when the input data slice is not an integral number of periods (see Figures 2, 3 and 4), the synthesis, that is, the output, is an integral number of periods. Thus, given non-periodic function, a Fourier synthesis will deliver up a periodic function.

Since the curves of Figures 1 and 5 are periodic as plotted, the Fourier coefficients are essentially zero except for the term which



I made the TRS-80* into a serious computer. Now I've made the Model II into a spectacular one.

I'm Irwin Taranto, and I've helped almost a thousand businesses get their first computers up and running.

I've done it primarily with the TRS-80, because it's a really elegant piece of hardware. Given the right programs, it can do substantially the same work as the traditional minicomputers that cost four times as much.

I proved it with four on-line, interactive programs adapted from the genuine Osborne & Associates systems, originally designed for the $30,000 Wang computer. Then I added two of my own and made them all work on a $4000 TRS-80.

Now I've done the same thing for the new TRS-80 Model II. It's an $8000 computer that works twice as fast and has four times the memory—up to two million characters.

My new systems are fully documented, and because I'm working with a much more powerful computer, they're a night-and-day advance over the Model I programs. They'll turn your Model II into a complete business computer, set up and ready to go.

THE TRS-80 MODEL II PROGRAMS

General Ledger/Cash Journal: handles up to 7000 transactions on 500 different user-defined accounts. It keeps track of them by month, quarter and year, makes comparisons to the prior year, and does departmentalization.

Accounts Payable/Purchase Order: generates the purchase order and posts the item to payables when the goods are received. Invoice-linked, it calculates and prints checks and aged ledger reports and links fully to the general ledger.

Accounts Receivable/Invoicing: keeps track of billed and unbilled invoices, open and closed items, aging and service charge calculation. It prints statements, links to the general ledger, and can work within either an invoice-linked or balance-forward accounting system.

Payroll/Job Costing: computes regular, overtime and piecework pay, keeps employee files, figures taxes and deductions, prints checks, journal, 941-A and W-2 forms, and breaks out individual job costs.

*A trademark of the Tandy Corporation.

When I say set up and ready to go, I mean just that. If you're not quite sure on that point, call the number below and we'll give you the names of some of the people who've already bought all over the world. Call them up and hear what they have to say.

These Model II programs are completely custom-tailored, which explains their $249.95 price. Before we'll send you a disk, you have to fill out a detailed questionnaire that tells us your precise business requirements. Then we send you the disk, all the instructions you need, and my phone number. If you call, we answer all your questions. If your questions are tough enough, I'll talk to you personally.

Because that way I'll make sure that Model II of yours turns into a spectacular computer, just like I promised.

☐ Please send me the custom questionnaires for the following $249.95 Model II programs:

- ☐ General Ledger/Cash Journal
- ☐ Accounts Payable/Purchase Order
- ☐ Accounts Receivable/Invoicing
- ☐ Payroll/Job Costing

☐ Please send me information on the TRS-80 Model I programs at $99.95 each

☐ Please send me information on other Taranto business programs

Your name _____

Company name _____

Address _____

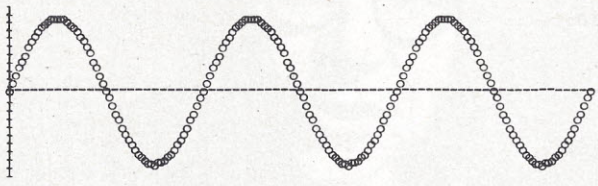
City/State/Zip _____

Taranto
& ASSOCIATES, INC.

P.O. Box 6073, 4136 Redwood Hwy., San Rafael CA 94903 • (415) 472-2670

```
Choose sinusoid, square, or triangle (1, 2, or 3): 1
State phase shift desired, in degrees: 0
What is the desired period? 180
What time increment will be used? 1
How many terms in Fourier series? 6

*********  INPUT DATA - TO BE SYNTHESIZED  *********
------ This curve is defined to be ONE PERIOD ------
```



N	A(N)	B(N)	C(N)
0	.00000006	.00000000	.00000006
1	.00000001	.00000002	.00000002
2	.00000007	-.00000009	.00000012
3	.00000014	.99999994	.99999990
4	-.00000007	-.00000001	.00000007
5	-.00000002	.00000003	.00000003
6	.00000000	.00000001	.00000001

```
######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```

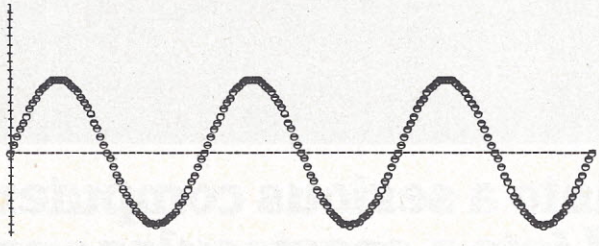
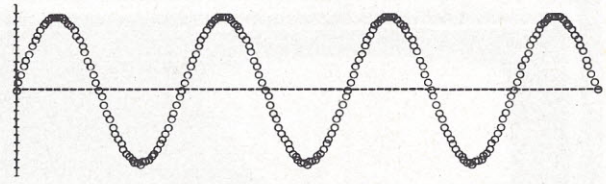


Figure 1.

```
Choose sinusoid, square, or triangle (1, 2, or 3): 1
State phase shift desired, in degrees: 0
What is the desired period? 210
What time increment will be used? 1
How many terms in Fourier series? 6

*********  INPUT DATA - TO BE SYNTHESIZED  *********
------ This curve is defined to be ONE PERIOD ------
```



N	A(N)	B(N)	C(N)
0	.18172519	.00000000	.18172519
1	.19789324	.00000001	.19789324
2	.26991479	.00000004	.26991479
3	.68542414	.00000001	.68542414
4	-.59434505	.00000006	.59434505
5	-.17492481	-.00000007	.17492481
6	-.09398414	-.00000005	.09398414

```
######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```

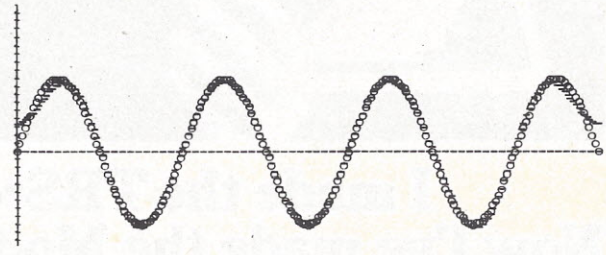
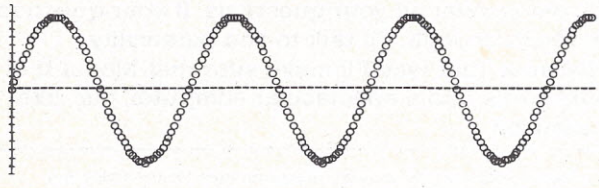


Figure 3.

```
Choose sinusoid, square, or triangle (1, 2, or 3): 1
State phase shift desired, in degrees: 0
What is the desired period? 195
What time increment will be used? 1
How many terms in Fourier series? 6

*********  INPUT DATA - TO BE SYNTHESIZED  *********
------ This curve is defined to be ONE PERIOD ------
```



N	A(N)	B(N)	C(N)
0	.09785204	.00000000	.09785204
1	.10809418	.03331488	.11311160
2	.15754967	.09706384	.18504942
3	.66199503	.61123774	.90102663
4	-.19034394	-.23404874	.30167802
5	-.07174375	-.11009936	.13141170
6	-.04075827	-.07491504	.08528482

```
######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```

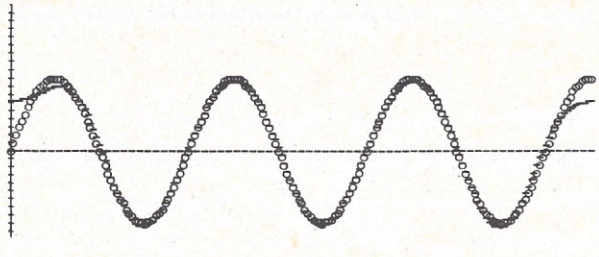
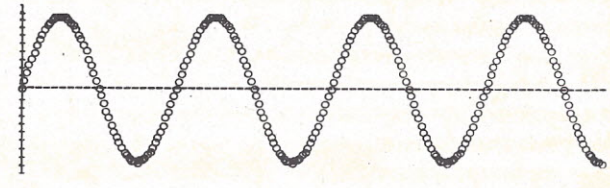


Figure 2.

```
Choose sinusoid, square, or triangle (1, 2, or 3): 1
State phase shift desired, in degrees: 0
What is the desired period? 225
What time increment will be used? 1
How many terms in Fourier series? 6

*********  INPUT DATA - TO BE SYNTHESIZED  *********
------ This curve is defined to be ONE PERIOD ------
```



N	A(N)	B(N)	C(N)
0	.08480512	.00000000	.08480512
1	.09130322	-.02438892	.09450449
2	.11854716	-.06330799	.13439245
3	.23570741	-.18869017	.30193039
4	-.61616111	.65707316	.90077725
5	-.10921252	.14540961	.18185525
6	-.05448962	.08693488	.10260015

```
######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```

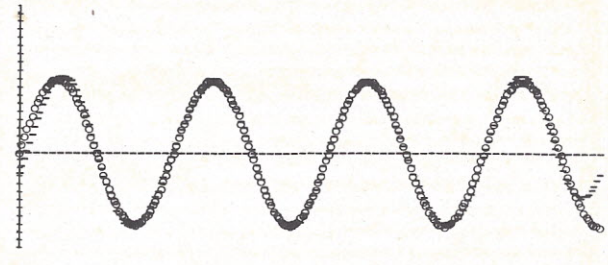


Figure 4.



How COMPUMAX stacks up to the competition

COMPARISON SHOPPING? STOP HERE!

IMPORTANT QUESTIONS ABOUT BUSINESS SOFTWARE	COMPUMAX	OSBORNE/ McGRAW HILL	PEACHTREE SOFTWARE	STRUCTURED SYSTEMS
What programs are available? *Are they INTERACTIVE?*	*INTERACTIVE:* GENERAL LEDGER ACC'TS PAYABLE ACC'TS REC'BLE INVENTORY PAYROLL/PERSONNEL	*INTERACTIVE:* GENERAL LEDGER ACC'TS PAYABLE ACC'TS REC'BLE *NON-INTERACTIVE:* PAYROLL	*INTERACTIVE:* GENERAL LEDGER ACC'TS PAYABLE ACC'TS REC'BLE PAYROLL *NON-INTERACTIVE:* INVENTORY	*INTERACTIVE:* GENERAL LEDGER ACC'TS PAYABLE ACC'TS REC'BLE *NON-INTERACTIVE:* STOCK CONTROL INVENTORY
What versions are available?	TRS-80, APPLE II COMMODORE PET MICROSOFT,CBASIC2 CP/M MICROPOLIS: EXIDY SORCERER, VECTOR MZ, DYNABYTE CROMEMCO III	WANG CBASIC2 CP/M®	MICROSOFT CP/M®	CBASIC2 CP/M®
What is the price?	MICROLEDGER, A/P, A/R, INV, PERS: $140.-each. MAXILEDGER, A/P, A/R: $350.-each.	GL, A/P, A/R, PAYROLL: $250.-each. Cost of configuring must be added!	GL, A/P, A/R, PAYROLL, INVENTORY $1000 each.	GL $995.- A/P $750.- A/R $750.- INV. $500.-
Hardware options	40 column CRT 64 column CRT 80 col. terminal 80 col. printer included	64 col. CRT only 132 column printer only	80 col. CRT only 132 column printer only	cursor addressable terminal only 132 column printer only.
Is source code included?	YES, INCLUDING PROGRAM FLOWCHARTS.	YES	YES	NO
What type of after-purchase support is offered?	1 YR WARRANTY & CORRECTION OF DEFECTS INDIVIDUAL PROGRAM AUTHORS AVAILABLE FOR QUESTIONS.	NONE	1 YR WARRANTY & CORRECTION OF DEFECTS THROUGH DEALER.	TECHNICIANS AVAILABLE FOR QUESTIONS. UPDATES MADE AVAILABLE FOR A FEE.

COMPUMAX is your one-stop shop for all your business bookkeeping software.

You've been led down the path before, but not this time. No more promises of turnkey computers without the key. It's YOUR turn to tell the computer how to run the business, not vice versa.

With COMPUMAX software you have a beginning. With 5 years of experience and over 3,000 systems installed, they are professionals, when it comes to solutions for the businessman.

COMPUMAX software is designed with CHANGE in mind, since everybody really wants his own touch added. The programs are SIMPLE, YET ELOQUENT.

COMPUMAX supplies ready, working programs. You can, then, easily customize them, as your additional requirements develop.

Or do as many have done—keep it simple by running the programs in their ready form. Join the microcomputer revolution the simple way.

For a demo, visit your local computer store. If you local retailer does not carry COMPUMAX software, have him give us a call at (415) 321-2881.

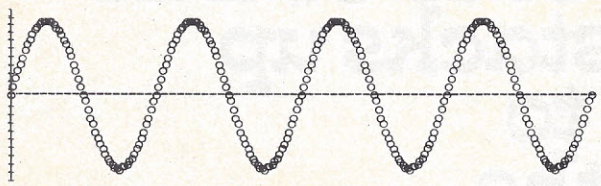
COMPUMAX

CP/M® is a registered trademark of Digital Research

467 HAMILTON AVENUE PALO ALTO, CA 94301

```
Choose sinusoid, square, or triangle (1, 2, or 3): 1
State phase shift desired, in degrees: 0
What is the desired period? 240
What time increment will be used? 1
How many terms in Fourier series? 6

*********  INPUT DATA - TO BE SYNTHESIZED  *********
------ This curve is defined to be ONE PERIOD ------
```



N	A(N)	B(N)	C(N)
0	.00000006	.00000000	.00000006
1	.00000000	-.00000000	.00000000
2	.00000001	-.00000008	.00000008
3	.00000000	-.00000002	.00000002
4	.00000029	1.00000010	1.00000010
5	-.00000007	.00000008	.00000011
6	-.00000003	.00000001	.00000004

```
######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```

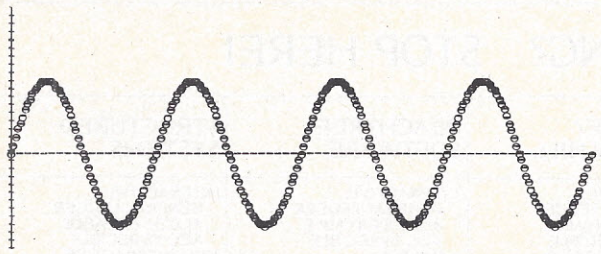


Figure 5.

states which sine wave in the series the input data represents. To illustrate, in Figure 1 we have chosen the period to be 180 seconds. The true period, however, is 60 seconds. The 60-second wave will appear as a third harmonic in a Fourier series that has a 180-second fundamental period. Therefore, all coefficients should equal zero except b_3, which should equal 1.0. Thus Figure 1 is correct. Figure 5 is also correct since a 60-second wave is a fourth harmonic in a series with a 240-second fundamental wave. Therefore b_4 is the only non-zero coefficient.

COMPUTATION OF TRUE PERIOD FROM FOURIER COEFFICIENTS

Now comes the big question. Given Fourier coefficient data for an arbitrarily assumed period, is it possible to extract the true period? Let's try it.

The assumed period in Figure 1 is 180 seconds. From the Fourier coefficients given in Figure 1 it is obvious that the only harmonic present is the third. Since 180 divided by three equals 60, the minimum period is pretty obviously 60 seconds. In Figure 2 the assumed period is 195 seconds. Note that all harmonics from the first to at least the sixth are present, and in both the cosine and sine components. If we assume that each column of Fourier coefficients represents a curve that first increases and then decreases as we go to higher harmonics, it is possible to determine the maximum value of the coefficient, and more to the point, the value of the harmonic at which the maximum occurs. This can be done as follows.

Let

$$y = ax^2 + bx + c$$

where y is the value of the Fourier coefficient and x is the harmonic number. Then, in the case of Figure 2, we can write this equation for values of x equal to 2, 3 and 4. Note that y can be interpreted as the value of either A(N), B(N) or C(N). Although C(N) would appear to be the more rational choice, we have solved the problem for all three choices and found little difference. Letting y be the value of C(N), the three equations, for Figure 2, are

$$.18505 = a(2)^2 + b(2) + c$$
$$.90103 = a(3)^2 + b(3) + c$$
$$.30168 = a(4)^2 + b(4) + c$$

These equations can be solved for a, b and c using the methods given in this column in the October and November 1979 issues. The matrix inversion method is discussed in the November column, but Program MATDEMO is given in the October column. Having

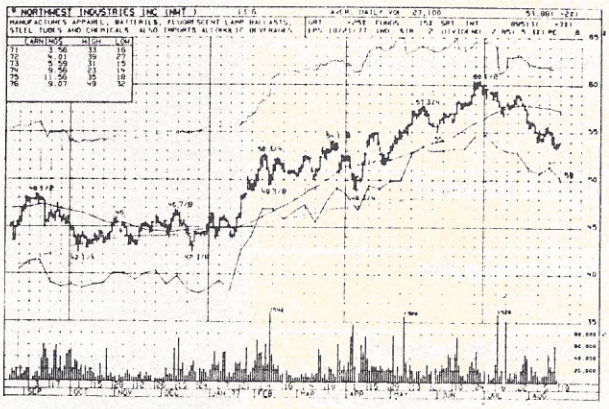


Figure 6.

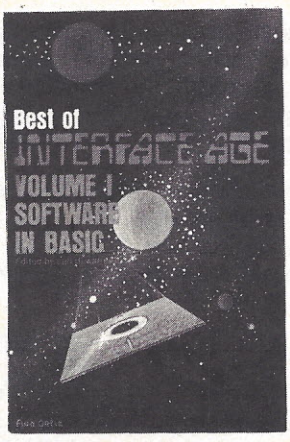
THE MARRIAGE OF THE CENTURY!

INTERFACE AGE, the country's leading personal computing magazine, has joined forces with DILITHIUM PRESS, one of America's fastest growing microcomputer publishers, to bring you "BEST OF INTERFACE AGE." These books will contain innovative and always informative articles on a specific theme, selected from the best issues of INTERFACE AGE.

VOLUME 1: SOFTWARE IN BASIC (2)

The five articles reprinted in this volume represent several different programming techniques. At the same time, they provide the reader with some of the most useful software programming ever created. Chapter 5, "The Great Experiment" offers the most requested piece of software to ever appear in the magazine, the complete source listing of Uiterwyck's 4K BASIC. The four other chapters — "Lawrence Livermore 8080 BASIC", "Dr. Wang's Palo Alto Tiny BASIC" and "National's Tiny BASIC" will give you an excellent grounding in the various phases of these popular language variations.

Available **400 pages $12.95**



VOLUME II: SOFTWARE (2)

This unique second volume presents thirteen system and applications software articles. The articles were chosen not only for their utility as working software systems but also for their value in showing broad spectrum of different programming techniques.

Forthcoming **150 pages** **$8.95**

To Order:

dilithium Press
P.O. Box 92
Dept. I.A.
Forest Grove, Or. 97116

solved for a, b and c, the value of x (the harmonic number) at which y becomes maximum can be found by differentiation to be —b/2a. If this value is divided into the assumed period, the result is the minimum period of the curve being analyzed.

Table 1 presents the results of these calculations for the curves of Figures 2, 3 and 4. Calculations were made using values of A(N), B(N) and C(N). In the case of Figure 3, the data at harmonics 2, 3 and 4, and at 3, 4 and 5 were used. Note that the results are fairly independent of which data were used. The concept of establishing the true period from an analysis of the Fourier coefficients seems to have some promise.

Table 1.

Assumed period	Harmonics used	Coefficient used	Indicated true period
195	2,3,4	A	64.62
		B	63.41
		C	64.09
210	2,3,4	A	63.17
		C	63.17
	3,4,5	A	65.10
		C	65.10
225	3,4,5	A	57.31
		B	56.55
		C	56.91

THE REAL WORLD

Having seen that we can analyze a function that is in fact periodic but whose period is not known, let us attempt to analyze a function that looks periodic but may in fact not be. What better example could we choose than the stock market? Figure 6 shows a chart of the price of the common stock of Northwest Industries, Inc. (NYSE symbol NWT) over the time period from September 1976 to August 1977, inclusive. The curve certainly looks as if it contained periodicities. Is this correct or is it only an illusion? There seems to be no reason why we should not do a Fourier analysis on this data, hoping that a half dozen or so terms turn out to be very large compared to the others. The presence of periodicities would be confirmed, the large coefficients would reveal their frequencies, and the real work could begin. The real world is ordinarily not so obliging, however, let us proceed optimistically.

We will need several new programs, as follows. First, a program to accept historical stock price information, prepare a data file, and provide some means for checking the data. Such a program is rather trivial to write, but is presented in Figure 7 for those unfamiliar with data files. Several improvements have already been implemented. First of all, the data was obviously read (slowly, painfully, and with much eye strain) from Figure 6. A major step forward will be taken when the new digitizer pad is installed. That will permit data to be picked off Figure 6 with a stylus and entered directly into RAM, from which the data file will be created on disk. Second, even without a digitizer, it was not necessary to enter the information into DATA statements. That was done to illustrate what could be done without a digitizer, a disk, or even data file capability. The second program required is essentially the front end of Program 4YESERES, presented last month. That is, a program that will plot the input data, in this case the data of Figure 6, or any part thereof, and then deter-

```
30 REM$$$$$$$$$$$$$$$$  P r o g r a m    S T O C K P R  $$$$$$$$$$$$$
40 REM
50 REM$$$$$$$$$$$$$$  Written by  -  Alfred A. Adler, Ph. D. $$$$$$$$$$
60 REM
70 REM$$$$$$$$$$$$$$$$  Version  1.0  -  November  1979  $$$$$$$$$$$$$
80 REM
85 DIM F(366)
90 CREATE "NWT",6,3
100 OPEN #1,"NWT"
105 REM Price of NWT from 9/76 to 8/77 inclusive.
108 REM Data 110  to 112 are for the quarter ended 9/30/76.
109 DATA 244 45.3,43.4,44.9,45.5,46.8,46.9,48.
110 DATA 47.75,47.85,47.9,48.5,47.9,48.,47.8,45.6,45.5
112 DATA 46.3,46.,46.5,46.65,46.,45.7,45.6,45.2,45.35
115 DATA 44.85,44.75,43.85,43.8,43.1,43.85,44.,43.725,42.7,43.35
118 DATA 43.2,43.5,43.85,43.6,44.5,44.25,43.25,43.6,44.,44.35,45.15
120 DATA 45.75,45.45,44.45,45.1,44.875,44.,43.15,43.9,43.5,43.25
122 DATA 43.5,44.5,45.25,45.25,45.1,44.875,45.5,46.,45.45,45.4,45.
125 DATA 45.7,44.3,44.7,45.85,46.3,46.25,46.2,45.6,45.3,45.25,44.65
128 DATA 44.25,43.,42.95,44.5,44.25,44.25,44.5,45.275
129 REM Data 115  to 128 are for the quarter ended 12/31/76.
130 DATA 46.,46.45,44.9,44.,44.7,45.8,45.95,45.8,45.75,45.1,44.15
132 DATA 44.,44.72,46.,46.,48.25,48.15,48.55,49.7,49.,49.6,49.4,50.5
135 DATA 51.,51.75,52.5,52.55,51.2,49.4,50.25,51.5,52.1,51.65,51.75
138 DATA 51.5,50.85,50.75,51.,50.75,50.75,50.5,51.25,51.25,52.35
140 DATA 52.5,52.4,52.25,50.85,50.35,50.5,51.,52.4,53.,53.,53.25
142 DATA 52.85,54.,53.25,53.7,53.35,51.5,52.55,52.,52.2
145 REM Data 130  to 142 are for the quarter ended 3/31/77.
148 DATA 52.5,51.25,50.5,49.1,50.35,50.,51.85,52.8,53.85,54.7,54.5
150 DATA 54.85,53.75,52.75,51.65,51.5,52.25,52.85,53.5,53.5,54.15
152 DATA54.3,55.,54.6,54.9,55.,55.8,55.85,57.,57.,56.9,57.35,57.45
155 DATA 57.75,56.85,56.8,55.9,55.75,55.6,55.65,56.4,56.4,56.95
158 DATA 56.95,56.7,57.,56.25,57.,58.25,58.15,57.4,58.,58.15,57.85
160 DATA 58.3,58.9,59.75,60.15,60.4,60.,59.35,59.25
162 REM Data 148  to 160 are for the quarter ended 6/30/77.
165 DATA 59.7,59.25,58.5,59.,58.4,57.75,57.15,57.75,57.9,57.6,57.7
168 DATA 58.45,58.8,58.5,57.85,57.45,56.2,55.4,56.15,55.9,55.1,54.55
170 DATA 54.9,54.25,54.9,55.25,55.1,55.15,54.4,54.,53.5,53.85
175 REM Data 165  to 170 are for the quarter ended 9/30/77.
200 DATA 4,41.5
400 READ D9
410 WRITE #1,D9
500 FOR D=1 TO D9
505 READ F(D)
510 WRITE #1,F(D)
515 NEXT
520 READ S8,S9
525 WRITE #1,S8,S9
530 CLOSE #1
570 OPEN #1,"NWT"
580 READ #1,D9
590 !D9
595 J=0
600 FOR D=1 TO D9
605 READ #1,F(D)
610 J=J+1
620 IF J<>1 THEN 700
625 !
630 !D,TAB(7),%7F3,F(D),
635 GOTO 800
700 !%7F3,F(D),
705 IF J=5 THEN J=0
800 NEXT
810 READ #1,S8,S9
820 !\!S8,S9
830 CLOSE #1
```

Figure 7.

THE HOT ONES!!!

Second printing now available. Customers nationwide have helped to make these our bestsellers! **Find out why.**

A STEP BY STEP INTRODUCTION TO 8080 MICROPROCESSOR SYSTEMS

David L. Cohn and James L. Melsa

This step by step presentation does not require any electronics or computer background. If you have access to an 8080 system, the book contains a number of exercises that illustrate the concepts discussed in each chapter; however, many of the exercises can be worked with pencil and paper.

$7.95

YOUR HOME COMPUTER

James White

A home computer can make life easier and more enjoyable. It can be simple to operate, as simple as a sewing machine. This book is written for the non-engineer, for the person who has no technical background. Its purpose is to introduce you to home computing and microcomputers.

$9.95

MY COMPUTER LIKES ME I SPEAK IN BASIC

Bob Albrecht

This book is about people, computers and a programming language called BASIC. It's an inch by inch, step by step guide to program your computer so it will do what you want it to do. Easy to read and a valuable tool for the computer hobbyist.

$3.95

To order:
dilithium Press
P.O. Box 92, Dept. IAB
Forest Grove, OR 97116

mine the Fourier coefficients and store them for future use, preferably as another data file. Finally, we need a program similar to the back end of Program 4YESERES. This program will pick up those Fourier coefficients chosen by the user as significant, disregarding the others, and reconstitute or synthesize the data from those coefficients only. The synthesized curve (the output) should be plotted and the input overlaid for comparison.

PROGRAMS STOCKPR, 4YEANAL, AND 4YESYNTH

Program STOCKPR, presented in Figure 7, has already been mentioned. This program CREATEs data file 'NWT' and then READs it back for verification.

Program 4YEANAL, presented in Figure 8, accepts the name of the data file to be analyzed in statement number 70. In this case the data file to be analyzed is 'NWT.' A file of Fourier coefficients is CREATEd which can be named by the user in statement number 160. The reader will recall that the first part of this month's column is devoted to a discussion of the choice of period of a quasi-periodic function. In keeping with those thoughts, several RUNs have been made on Program 4YEANAL, choosing periods of 241, 210 and 180 seconds. These runs are shown in Figure 9, 10 and 11. The corresponding coefficient data files are named 'CONWT,' 'CONWT210,' and 'CONWT180,' respectively.

Program 4YESYNTH is presented in Figure 12. This program accepts the name of the coefficient file, from which it will synthesize the output, in statement number 70. It accepts the name of the original data file, from which it will overlay the input data, in statement number 68. Both 4YESYNTH and 4YEANAL are not too different from 4YESERES, and both are well annotated so further remarks should be unnecessary.

```
LIST

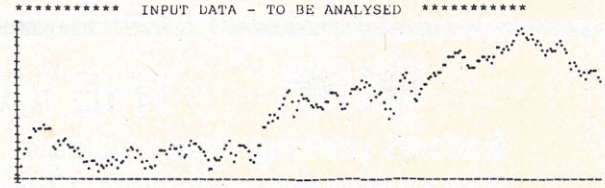
5 REM4444444444444444444444444444444444444444444444444444444444444444444444
10 REM
15 REM444444444444444  P r o g r a m    4 Y E A N A L  444444444444444
20 REM
25 REM444444444444  Written by  -  Alfred A. Adler, Ph.D.  44444444444
30 REM
35 REM44444444444444  Version  2.0  -  December  1979  444444444444444
40 REM
45  DIM F(366),A(100),B(100),C(100)
50  P2=6.2831853
55 !"444444  Program  4YEANAL  -  by  Alfred  A. Adler  Ph.D.  444444"
60 !
65        REM ****** SELECT AND READ DATA
70 INPUT"Choose data file name : ",W$
75 OPEN #1,W$
80 READ #1,D9
85 FOR D=1 TO D9
90 READ #1,F(D)
95 NEXT
100 REM S8 is slightly less than 80 divided by (Ymax.-Ymin.).
105 REM   It is the scaling factor for Y, see lines 790 and 1193.
110 REM S9 is slightly less than Ymin.
115 REM   It is to keep the ordinate positive, therefore on the grid.
120 READ #1,S8,S9
125 CLOSE #1
130       REM ****** INPUT DATA
135 INPUT"What is the initial time, (must be 1 or more)? ",T0
140 INPUT"What is the assumed period? ",T9
145 INPUT"Choose a time increment between data points. ",T1
150 INPUT"How many terms in Fourier series? ",N1
155 N2=INT(N1/25+1)
160 INPUT"Choose name for coefficient file: ",C$
162       REM ****** CREATE NEW FILE
165 CREATE C$,N2,3
170 OPEN #1,C$
175 WRITE #1,N1,S8,S9
182       REM ****** JUMP DOWN TO PLOT ROUTINES
185 GOSUB 305
190   !" N",TAB(16),"A(N)",TAB(34),"B(N)",TAB(52),"C(N)"
195       REM ****** COMPUTATION OF FOURIER COEFFICIENTS
200 FOR N=0 TO N1
205 A=0\B=0\A(N)=0\B(N)=0
210 FOR T=T0 TO T0+T9 STEP T1
215 A=F(T)*COS(P2*N*(T-T0)/T9)
220 IF T=T0 THEN A=A/2
225 IF T=T0+T9 THEN A=A/2
230 A(N)=A(N)+A
235 B=F(T)*SIN(P2*N*(T-T0)/T9)
240 IF T=T0 THEN B=B/2
245 IF T=T0+T9 THEN B=B/2
250 B(N)=B(N)+B
255 NEXT
260 A(N)=A(N)*2*T1/T9
265 B(N)=B(N)*2*T1/T9
270 C(N)=SQRT(A(N)^2+B(N)^2)
275 WRITE #1,A(N),B(N),C(N)
280 !N,TAB(10),%12F8, A(N),TAB(28),B(N),TAB(46),C(N)
285 NEXT
290 CLOSE #1
295 !\!
300 END
302       REM ****** PLOTTING ROUTINES
305 !
```

Figure 8a.

```
310 !"***********  INPUT DATA - TO BE ANALYSED  ***********"
315 !
320       REM ****** PLOT AXES
325 P=3.1415927
330 A0=4102\REM 1006n
335 A1=4112\REM 1010n
340 X8=4440\REM1158n
345 X9=4441\REM1159n
350 Y8=4438\REM1156n
355 Y9=4439\REM 1157n
360 C0=CALL(A0)
365 REM C0 not required to run but it does clear X and Y.
370 C$="-"
375 FILL 4442,ASC(C$)
380 R=80
385 FILL Y8,R
390 FILL Y9,0
395 REM This moves Y down to +80, where we want the X axis.
400 FOR X=0 TO 360 STEP 5
405 REM This plots X axis at 12 to the inch.
410 GOSUB 495
415 C1=CALL(A1)
420 NEXT
425 C$="1"
430 FILL 4442,ASC(C$)
435 FILL X8,0
440 FILL X9,0
445 REM This puts X at 0, where we want Y axis.
450 U=80
455 FOR Y=0 TO U   STEP 4
460 REM This plots Y axis at 12 to the inch.
465 GOSUB 530
470 C1=CALL(A1)
475 NEXT
478       REM ****** JUMP DOWN AND PLOT INPUT DATA
480 GOTO 560
490       REM ****** SUBROUTINE TO PLOT X (HORIZONTAL)
495 X0=X/256
500 X2=INT(X0)
505 X1=INT((X0-X2)*256)
510 FILL X8,X1
515 FILL X9,X2
520 RETURN
525       REM ****** SUBROUTINE TO PLOT Y (VERTICAL)
530 Y0=Y/256
535 Y2=INT(Y0)
540 Y1=INT((Y0-Y2)*256)
545 FILL Y8,Y1
550 FILL Y9,Y2
555 RETURN
560       REM ****** COMPUTE AND SCALE INPUT DATA
565 C$="."
570 FILL 4442,ASC(C$)
575 FOR T=T0 TO T0+T9 STEP T1
580 X=360*(T-T0)/T9
585 GOSUB 495
590 S1=80
600 Y=S1-S8*(F(1)-S9)
605 GOSUB 530
610 C1=CALL(A1)
615 NEXT
618       REM ****** SKIP BELOW GRAPH TO CONTINUE
620 FILL 4442,95
625 S=96
630 S0=S/256
635 S2=INT(S0)
640 S1=INT((S0-S2)*256)
645 FILL X8,0
650 FILL X9,0
655 FILL Y8,S1
660 FILL Y9,S2
665 C1=CALL(A1)
668       REM ****** GO BACK AND COMPUTE FOURIER COEFFICIENTS
670 RETURN
READY
BYE
```

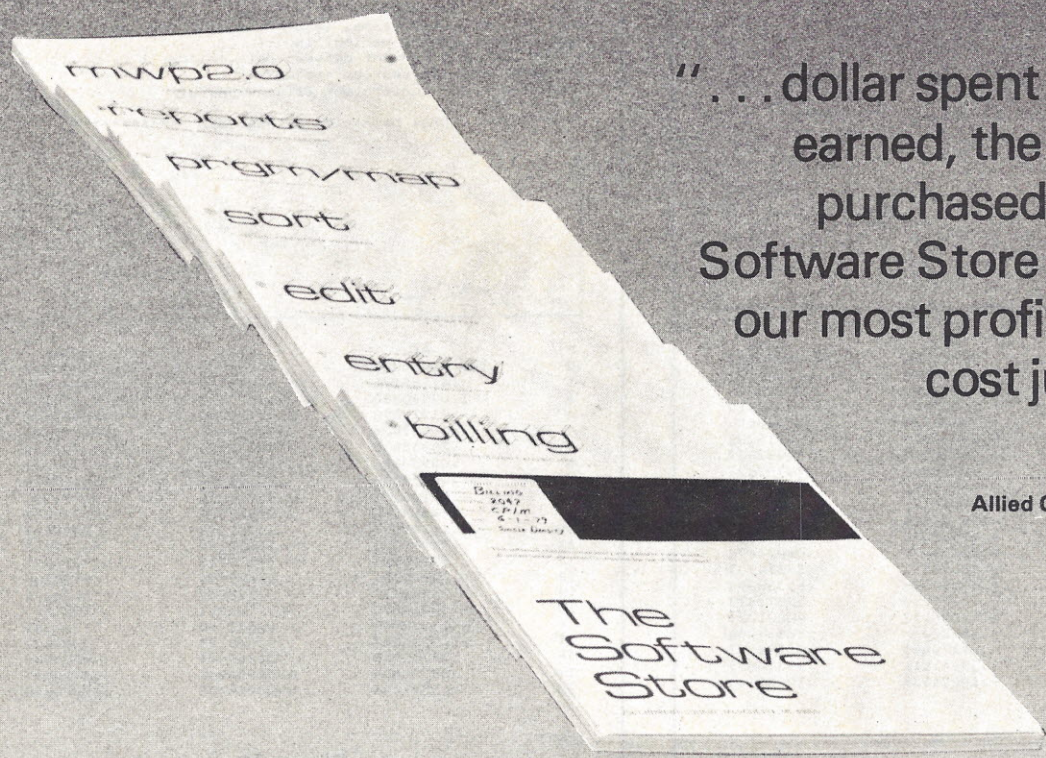
Figure 8b.

```
Choose data file name : NWT
What is the initial time, (must be 1 or more)? 1
What is the assumed period? 241
Choose a time increment between data points. 1
How many terms in Fourier series? 25
Choose name for coefficient file: CONWT
***********  INPUT DATA - TO BE ANALYSED  ***********
```



N	A(N)	B(N)	C(N)
0	100.98689000	.00000000	100.98689000
1	1.03070220	-6.51342030	6.59446670
2	.64502473	-2.10578720	2.20236160
3	-.79680008	-1.23680270	1.47124820
4	.44099834	-.38474490	.58524199
5	-.45058315	-.09189949	.45985943
6	-.31959632	-.36717267	.48678289
7	.04680933	.24593071	.25034581
8	-.08983769	-.49803307	.50607089
9	-.08113851	-.29316647	.30418750
10	-.14973700	-.58009585	.59910965
11	-.00297141	.03758919	.03770645
12	-.01458369	.06464575	.06627033
13	-.00348088	-.46001502	.46002819
14	-.17166607	.08535252	.19171409
15	.23599249	-.23307973	.33169055
16	.17046325	-.36345428	.40144331
17	.00165074	-.03689823	.03693514
18	.05594095	-.52710145	.53006163
19	.00417942	-.38918754	.38920998
20	.15953633	-.08329052	.17996986
21	.13121900	-.13054201	.18509361
22	.00789839	-.37656970	.37665252
23	-.04553159	-.06179078	.07675432
24	.27693179	-.04637378	.28078772
25	-.05293048	-.33315209	.33733064

Figure 9.



"...dollar spent for dollar earned, the software purchased from the Software Store has been our most profitable and cost justified."

George Brown
Allied Computer Services
Huntington WV

Why reinvent the wheel? The Software Store supplies complete program systems written in easy to use Microsoft BASIC for Radio Shack Model II, Altair/MITS, TEI, Cromemco, North Star, Processor Tech, Altos, Ohio Scientific, Billings, IMSAI, Digital Micro Systems and other Z80 & 8080 based computers. Our growing family of products is divided into three categories: *application utilities, systems* and *system utilities.*

The *application utilities* are the basic building blocks for application program systems. Almost every application can be made of a key-to-disk data entry segment, a file edit segment, a sort/merge segment, a record selection segment and a report & file update segment. These functions are carried out by the ENTRY, EDIT, SORT, SELECT and REPORTS systems, respectively. Application utilities consist of two programs: one for interactive task definition and the other for task execution. Once defined, a task may be executed any number of times or easily revised.

Application utilities permit rapid solutions to satisfy each user's unique requirements. Many first time computer users have built respectable application systems using our utilities and self instructive documentation. Computer stores and consultants utilize our products to generate custom systems for their clients. Because of the flexible and interactive design of the task definition programs, previously defined systems can be easily revised to meet changing needs.

The *systems* are complete packages for a specific application. Systems are fabricated from application utilities together with application specific programs. For example, our Accounts Receivable System utilizes the ENTRY, EDIT, SORT, SELECT and MWP systems along with six special billing system programs.

The MWP system is a complete word processing system with flexible user defined "name & address" files. The "name and address" information and date can be inserted throughout a document. The documents might be reports, manuals, mailing labels, letters or legal documents.

The *system utilities* include programming tools such as the Program Map BASIC cross reference program along with general utilities such as the Disk Fix file recovery program, the Disk Copy (1D & 2D) diskette copy program, the TX-RX file transfer and media conversion programs and the CATALOG diskette library index program.

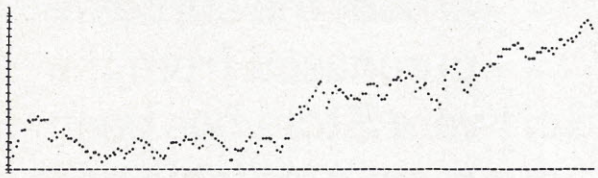
To find out more about our growing family of software products, contact your local computer dealer for a demonstration or contact us.

The Software Store

706 Chippewa Square
Marquette MI 49855
(906) 228-7622

```
Choose data file name : NWT
What is the initial time, (must be 1 or more)? 1
What is the assumed period? 210
Choose a time increment between data points. 1
How many terms in Fourier series? 20
Choose name for coefficient file: CONWT210

***********  INPUT DATA - TO BE ANALYSED  ***********
```



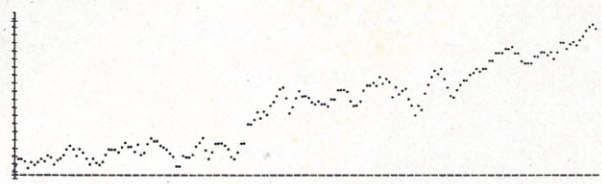
N	A(N)	B(N)	C(N)
0	99.08566700	.00000000	99.08566700
1	1.98658850	-5.47184760	5.82131000
2	1.16865210	-1.34413680	1.78113760
3	.57014824	-1.90992500	1.99320910
4	.16509640	-.20688630	.26468616
5	-.13728104	-.64982024	.66416295
6	.36384766	-.16600718	.39992937
7	.01956000	-.79729833	.79753823
8	.01185689	-.48564276	.48578748
9	.36580079	-.77896171	.86057630
10	.03779199	-.39894606	.40073207
11	-.32990957	-.49359095	.59369382
12	.08474031	-.02636417	.08874678
13	.22041733	-.49597433	.54274703
14	.18645517	-.49994629	.53358394
15	-.16513251	-.17246686	.23877513
16	.23386570	-.43094192	.49031022
17	.37420640	-.45402013	.58835764
18	.15917330	-.31170565	.34999507
19	-.03915107	-.39141511	.39336827
20	.06290626	-.10221519	.12002143

```
READY
BYE
+
```

Figure 10.

```
Choose data file name : NWT
What is the initial time, (must be 1 or more)? 30
What is the assumed period? 180
Choose a time increment between data points. 1
How many terms in Fourier series? 20
Choose name for coefficient file: CONWT180

***********  INPUT DATA - TO BE ANALYSED  ***********
```



N	A(N)	B(N)	C(N)
0	100.08078000	.00000000	100.08078000
1	.41766116	-5.46833820	5.48426520
2	1.09990390	-2.14970530	2.41475080
3	-.71119578	-1.74139390	1.88102420
4	.20507434	-1.38305060	1.39817180
5	.29335839	-.42580270	.51707551
6	.08950888	-1.10877600	1.11238310
7	.06954516	-.58706117	.59116610
8	-.00208723	-1.08794010	1.08794220
9	.04063050	-.86695722	.86790879
10	-.14316997	-.11792349	.18548205
11	.34273073	-.53013822	.63127718
12	.19224787	-.56199717	.59396975
13	-.08746794	-.03498993	.09420688
14	.06963337	-.33928941	.34636124
15	-.02114623	-.57882961	.57921575
16	-.03170902	-.57620178	.57707361
17	-.11284264	-.17549754	.20864527
18	.24733151	-.08142474	.26038983
19	.02976076	-.63358239	.63428097
20	.15571026	-.08626302	.17800840

```
READY
```

Figure 11.

```
LIST

5 REM4444444444444444444444444444444444444444444444444444444444444444444444
10 REM
15 REM44444444444444  P r o g r a m   4 Y E S Y N T H  44444444444444
20 REM
25 REM444444444444  Written by  -  Alfred A. Adler, Ph.D.  44444444444
30 REM
35 REM44444444444444  Version  2.0  -  December  1979  444444444444444
40 REM
45  DIM F(366),A(100),B(100),C(100),N(12)
50  P2=6.2831853
55 !"444444  Program  4YESYNTH  -  by  Alfred  A. Adler  Ph.D.  444444"
60 !
65        REM ****** SELECT AND READ COEFFICIENT DATA
68 INPUT"Choose data file: ",W$
70 INPUT"Choose coefficient file: ",C$
75 OPEN #1,C$
78 OPEN #2,W$
80 READ #1,N1,S8,S9
82 READ #2,D9
85 FOR N=0 TO N1
90 READ #1,A(N),B(N),C(N)
93 NEXT
94 FOR D=1 TO D9
95 READ #2,F(D)
98 NEXT
99 CLOSE #2
100 REM S8 is slightly less than 80 divided by (Ymax.-Ymin.).
105 REM   It is the scaling factor for Y, see lines 565 and 700.
108 REM S9 is slightly less than Ymin.
110 REM   It is to keep the ordinate positive, therefore on the grid.
120 CLOSE #1
122       REM ****** INPUT DATA
125 INPUT"What is the initial time, (must be 1 or more)? ",T0
130 INPUT"What is the assumed period? ",T9
135 INPUT"Choose a time increment between data points. ",T1
140 INPUT"How many terms in Fourier series will be used (max. 12)? ",G
145 !"State term numbers: ",
150 ON G GOTO 155,160,165,170,175,180,185,190,195,200,202,203
155 INPUT N(1)\GOTO205
160 INPUT N(1),N(2)\GOTO205
165 INPUT N(1),N(2),N(3)\GOTO205
170 INPUT N(1),N(2),N(3),N(4)\GOTO205
175 INPUT N(1),N(2),N(3),N(4),N(5)\GOTO205
180 INPUT N(1),N(2),N(3),N(4),N(5),N(6)\GOTO205
185 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7)\GOTO205
190 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7),N(8)\GOTO205
195 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7),N(8),N(9)\GOTO205
200 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7),N(8),N(9),N(10)\GOTO205
202 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7),N(8),N(9),N(10),N(11)\GOTO205
203 INPUT N(1),N(2),N(3),N(4),N(5),N(6),N(7),N(8),N(9),N(10),N(11),N(12)
204 GOTO 205
205 W=0\!
212       REM ****** PRINT FOURIER COEFFICIENTS
215   !" N",TAB(16),"A(N)",TAB(34),"B(N)",TAB(52),"C(N)"
220 FOR M=1 TO G
225 N=N(M)
230 !N,TAB(10),%12F8, A(N),TAB(26),B(N),TAB(46),C(N)
235 NEXT
238       REM ****** JUMP DOWN TO PLOT ROUTINES
240 GOSUB 265
245 IF W<>1 THEN 260
250 !"*****  ONE OR MORE POINTS WERE OUT OF BOUNDS  *******"
255 !
```

Figure 12a.

MICRO AID®

THE TEACHER'S HELPER

MICRO AID® is an educational computer package designed to free the teacher from much of the drugery inherent in testing, tracking, and grading students.

FEATURING: ■ Quiz with instruction mode ■ Computer aided instruction with quiz mode ■ Multi student participation ■ Recording of student grades ■ 3 levels of subject knowledge for individualized testing scores ■ Inventory build-up of test questions ■ Analysis of answers per student and entire class ■ Randomized questions rarely duplicating a quiz ■ Hard copy printed questions for major testing.

NO TECHNICAL EXPERIENCE NEEDED: Teacher simply types in questions, answers, and explanations. MICRO AID® is designed as a plug-in, easy to use system with no technical background required.

TEACHER IN FULL COMMAND: MICRO AID® is "menu-driven" in that it allows the teacher to select individualized teaching functions, perform an activity within a function, choose the number of questions to be on a quiz and sign on each student as he/she sits at the terminal.

INEXPENSIVE & COST EFFECTIVE: MICRO AID® system—$9,900, includes 4 CRT keyboard terminals, CPU & printer, software, and instruction manuals. Software only, $750. Instruction manual only, $40. Write or call for detailed information.

MICRO INNOVATIONS CORP.
420 LEXINGTON AVE., NY 10017
(212) 687-3637 *DEALER INQUIRIES INVITED*

CIRCLE INQUIRY NO. 48

```
262       REM ****** PLOTTING ROUTINES
265 !
270 !"######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######"
275 !
280       REM ****** PLOT AXES
285 P=3.1415927
290 A0=4102\REM 1006H
295 A1=4112\REM 1010H
300 X8=4440\REM1158H
305 X9=4441\REM1159H
310 Y8=4438\REM1156H
315 Y9=4439\REM 1157H
320 C0=CALL(A0)
325 REM C0 not required to run but it does clear X and Y.
330 C$="-"
335 FILL 4442,ASC(C$)
340 R=140
350 FILL Y8,R
355 FILL Y9,0
360 REM This moves Y down to +80, where we want the X axis.
365 FOR X=0 TO 360 STEP 5
370 REM This plots X axis at 12 to the inch.
375 GOSUB 460
380 C1=CALL(A1)
385 NEXT
390 C$="1"
395 FILL 4442,ASC(C$)
400 FILL X8,0
405 FILL X9,0
410 REM This puts X at 0, where we want Y axis.
415 U=140
420 FOR Y=0 TO U   STEP 4
425 REM This plots Y axis at 12 to the inch.
430 GOSUB 495
435 C1=CALL(A1)
440 NEXT
442       REM ****** JUMP DOWN AND PLOT OUTPUT DATA
445 GOTO 640
455       REM ****** SUBROUTINE TO PLOT X (HORIZONTAL)
460 X0=X/256
465 X2=INT(X0)
470 X1=INT((X0-X2)*256)
475 FILL X8,X1
480 FILL X9,X2
485 RETURN
490       REM ****** SUBROUTINE TO PLOT Y (VERTICAL)
495 Y0=Y/256
500 Y2=INT(Y0)
505 Y1=INT((Y0-Y2)*256)
510 FILL Y8,Y1
515 FILL Y9,Y2
520 RETURN
530       REM ****** COMPUTE AND SCALE INPUT DATA
535 C$="."
540 FILL 4442,ASC(C$)
545 FOR T=T0 TO T0+T9 STEP T1
550 X=360*(T-T0)/T9
555 GOSUB 460
560 S1=110
565 Y=S1-S8*(F(T)-S9)
570 GOSUB 495
575 C1=CALL(A1)
580 NEXT
582       REM ****** SKIP BELOW GRAPH TO CONTINUE
585 FILL 4442,95
590 S=160
595 S0=S/256
600 S2=INT(S0)
605 S1=INT((S0-S2)*256)
610 FILL X8,0
615 FILL X9,0
620 FILL Y8,S1
625 FILL Y9,S2
630 C1=CALL(A1)
635 RETURN
640       REM ****** COMPUTE AND SCALE OUTPUT DATA
645 C$="-"
650 FILL 4442,ASC(C$)
655 FOR T=T0 TO T0+T9
660 X=360*(T-T0)/T9
665 GOSUB 460
670 Y=0
675 FOR M=1 TO G
680 N=N(M)
682 IF N<>0 THEN 685
683 Y=A(0)/2\GOTO 695
685 W=A(N)*COS(P2*N*(T-T0)/T9)+B(N)*SIN(P2*N*(T-T0)/T9)
690 Y=Y+W
695 NEXT
700 Y=110-S8*(Y-S9)
705 IF Y<0 THEN 725
710 GOSUB 495
715 C1=CALL(A1)
720 GOTO 730
725 W=1
730 NEXT
732       REM ****** GO BACK AND OVERPLOT INPUT DATA
735 GOTO 535
READY
```

Figure 12b.

PARDON US WHILE WE SHAKE THINGS UP.

We've never been interested in doing another me-too CPU board, so we waited until we had a product that was the full equal of our memories and other peripherals. Now we have **two** CPU boards: a Z-80 version (and we all know how powerful that chip can be), along with our amazing 8085/8088 dual processor board. The 8088 acts like an 8086 hidden in an 8 bit package; it has 16 bit internal operation but works with an 8 bit bus, can execute all 8086 code, and runs at 5 MHz while the 8085 can run at a slower speed (if needed) for compatibility with the rest of a system . . . it's almost like having a 16 pit power with an 8 bit bus. Both CPU boards — in fact, **all** of our S-100 boards — meet all IEEE S-100 bus specifications. Full information and pricing on these advanced products, including details on the ground-breaking 8085/8088 board, will be available from us starting **April 2nd** (sorry, no additional information will be given out before that date).

LOOKING FOR MEMORY?

Then look for a board that's static, runs up to 5 MHz, meets the IEEE S-100 standards, is low in power, includes a 1 year warranty, and has the **CompuPro** name. Choose from **unkit** (sockets, bypass caps pre-soldered in place for easy assembly, **assembled**, and boards qualified under our high-reliability **Certified System Component** (CSC) program.

Memory Name	Buss & Notes	Unkit	Assm	CSC
8K Econoram* IIA	S-100	**$169**	**$189**	**$239**
16K Econoram XIV	S-100 (1)	**$299**	**$349**	**$429**
16K Econoram X-16	S-100	**$329**	**$379**	**$479**
16K Econoram XIIIA-16	S-100 (2)	**$349**	**$419**	**$519**
16K Econoram XV-16	H8 (3)	**$339**	**$399**	n/a
16K Econoram IX-16	Dig Grp	**$319**	**$379**	n/a
24K Econoram VIIA-24	S-100	**$449**	**$499**	**$599**
24K Econoram XIIIA-24	S-100 (2)	**$479**	**$539**	**$649**
32K Econoram X-32	S-100	**$599**	**$689**	**$789**
32K Econoram XIIIA-32	S-100 (2)	**$649**	**$729**	**$849**
32K Econoram XV-32	H8 (3)	**$649**	**$749**	n/a
32K Econoram IX-32	Dig Grp	**$599**	**$679**	n/a
32K Econoram XI	SBC/BLC	n/a	n/a	**$1050**

***Econoram** is a trademark of **Godbout Electronics.**

(1) Extended addressing (24 address lines). Addressable on 4K boundaries.
(2) Compatible with all bank select systems (Cromemco, Alpha Micro, etc.); addressable on 4K boundaries.
(3) Bank select option for implementing memory systems greater than 64K.

OTHER S-100 BUSS PRODUCTS

2S "Interfacer I" I/O board.....**$199** unkit, **$249** assm, **$324** CSC
3P Plus S "Interfacer II" I/O board.....**$199** unkit, **$249** assm, **$324** CSC
Godbout computer enclosure.....**$329** rack mount, **$289** desktop
6 slot high performance motherboard.....**$89** unkit, **$129** assm
12 slot high performance motherboard....**$129** unkit, **$169** assm
19 slot high performance motherboard....**$174** unkit, **$214** assm
2708 EROM board (less EROMs).....**$85** unkit
Active terminator board.....**$34.50** kit
Memory Manager board.....**$59** unkit, **$85** assm, **$100** CSC
PASCAL/M™ 8″ diskette.....**$350** (Z-80 or 8080/8085 version)
Special!!! Above diskette costs only $150 when purchased with any CompuPro memory board.

MEMORY EXPANSION KIT.

Expands memory in **TRS-80** -I** and **-II, Heath H89, Apple, Exidy,** and other machines. **$87.20** for 8 low power, high speed (250 ns/4 MHz), 16K dynamic RAMs. Add $3 for dip shunts and installation instructions for the TRS-80**.

**TRS-80 is a trademark of the Tandy Corporation.

TERMS: Cal res add tax. Allow 5% for shipping, excess refunded. VISA®/Mastercharge® call our 24 hour order desk at **(415) 562-0636.** COD OK with street address for UPS. Sale prices good through cover month of magazine; other prices are subject to change without notice.

CompuPro™ from **GODBOUT ELECTRONICS**

Bldg. 725, Oakland Airport, CA 94614

FINALLY, THE RESULTS . . .

A careful examination of Figures 9, 10 and 11 reveals that our hopes have not been completely in vain. There are a reasonably small number of Fourier coefficients whose values are very significantly greater than those of their fellows. There are also, unfortunately, a rather complete spectrum of values. Separating those coefficients large enough to be 'significant' from those small enough to be 'insignificant ' will not only be difficult but also to some extent arbitrary. There certainly are, however, many values that clearly are insignificant, so the list of harmonics present in the input can certainly be truncated to some extent. In any event, let us start by including as few as can even be remotely justified and observe the results.

We certainly cannot hope to obtain any kind of synthesis of the input data of Figure 9 if at least the first four terms of data file CONWT are not included. A synthesis using only these terms is shown in Figure 13, and it is surprisingly accurate. Note first of all that it follows the mean line of the data quite well, leaving out the higher frequency terms. This of course is to be expected, since we just eliminated the coefficients of those terms. Note further that the synthesis departs from the input data at the two end points, just as occurred in Figures 2, 3 and 4.

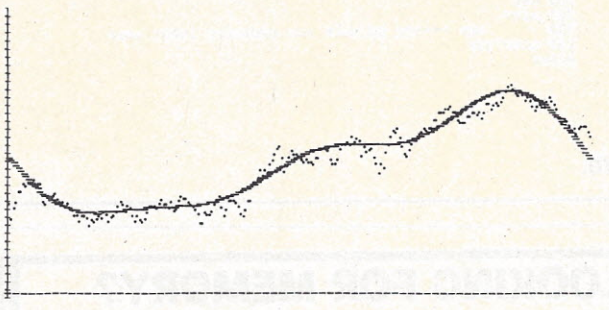
Flush with this success and in a more generous frame of mind, we next select the 11 largest Fourier coefficients from among the first 20. From this we obtain a very good synthesis, see Figure 14. The reader should bear in mind that this output is the result of synthesizing a curve using only the 11 Fourier coefficients shown in the figure. There is no other direct computational relationship between the output and the input. With that in mind, there is excellent agreement between the input and the output data. Regarding the use of more terms, we have certainly already reached the point of diminishing returns.

Choosing the 11 largest coefficients in Figure 10, and the eight largest in Figure 11, we obtain the syntheses shown in Figures 15

```
Choose data file: NWT
Choose coefficient file: CONWT
What is the initial time, (must be 1 or more)? 1
What is the assumed period? 241
Choose a time increment between data points. 1
How many terms in Fourier series will be used (max. 12)? 4
State term numbers: ?0,1,2,3

 N          A(N)           B(N)           C(N)
 0     100.98689000     .00000000    100.98689000
 1       1.03070220   -6.51342030      6.59446670
 2        .64502473   -2.10578720      2.20236160
 3       -.79680008   -1.23680270      1.47124820

######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```



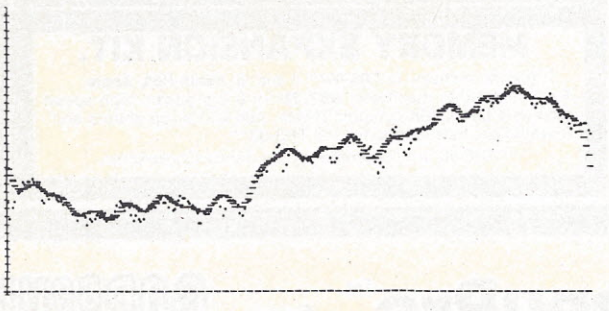
```
READY
```

Figure 13.

```
Choose data file: NWT
Choose coefficient file: CONWT
What is the initial time, (must be 1 or more)? 1
What is the assumed period? 241
Choose a time increment between data points. 1
How many terms in Fourier series will be used (max. 12)? 11
State term numbers: ?0,1,2,3,4,5,6,8,10,13,18

 N          A(N)           B(N)           C(N)
 0     100.98689000     .00000000    100.98689000
 1       1.03070220   -6.51342030      6.59446670
 2        .64502473   -2.10578720      2.20236160
 3       -.79680008   -1.23680270      1.47124820
 4        .44099834    -.38474490       .58524199
 5       -.45058315    -.09189949       .45985943
 6       -.31959632    -.36717267       .48678289
 8       -.08983769    -.49803307       .50607089
10       -.14973700    -.58009585       .59910965
13       -.00348088    -.46001502       .46002819
18        .05594095    -.52710145       .53006163

######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```



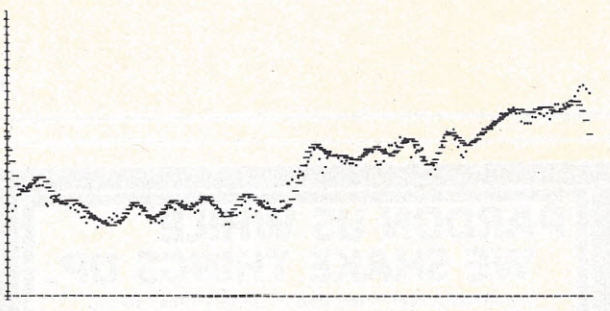
```
READY
```

Figure 14.

```
Choose data file: NWT
Choose coefficient file: CONWT210
What is the initial time, (must be 1 or more)? 1
What is the assumed period? 210
Choose a time increment between data points. 1
How many terms in Fourier series will be used (max. 12)? 11
State term numbers: ?0,1,2,3,5,7,9,11,13,14,17

 N          A(N)           B(N)           C(N)
 0     99.08566700     .00000000     99.08566700
 1      1.98658850   -5.47184760      5.82131000
 2      1.16865210   -1.34413680      1.78113760
 3       .57014824   -1.90992500      1.99320910
 5      -.13728104    -.64982024       .66416295
 7       .01956000    -.79729833       .79753823
 9       .36580079    -.77896171       .86057630
11      -.32990957    -.49359095       .59369382
13       .22041733    -.49597433       .54274703
14       .18645517    -.49994629       .53358394
17       .37420640    -.45402013       .58835764

######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```



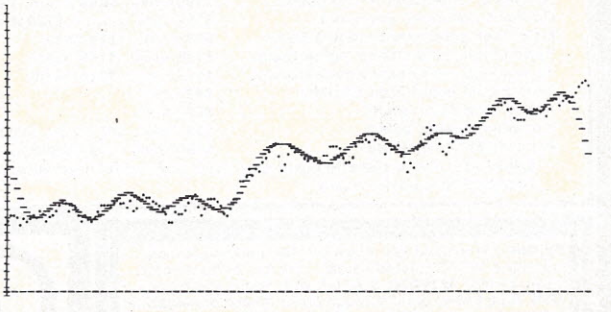
```
READY
```

Figure 15.

```
Choose data file: NWT
Choose coefficient file: CONWT180
What is the initial time, (must be 1 or more)? 30
What is the assumed period? 180
Choose a time increment between data points. 1
How many terms in Fourier series will be used (max. 12)? 8
State term numbers: ?0,1,2,3,4,6,8,9

 N          A(N)           B(N)           C(N)
 0     100.08078000     .00000000    100.08078000
 1        .41766116   -5.46833820      5.48426520
 2       1.09990390   -2.14970530      2.41475080
 3       -.71119578   -1.74139390      1.88102420
 4        .20507434   -1.38305060      1.39817180
 6        .08950888   -1.10877600      1.11238310
 8       -.00208723   -1.08794010      1.08794220
 9        .04063050    -.86695722       .86790879

######  OUTPUT DATA - SYNTHESIS OF INPUT DATA  ######
```



```
READY
```

Figure 16.

The Fifth
Trenton Computer Festival
TCF-80

April 19 & 20, 1980

10 AM to 6 PM SATURDAY, 19th

at TRENTON STATE COLLEGE Trenton, New Jersey

10 AM to 4 PM SUNDAY, 20th



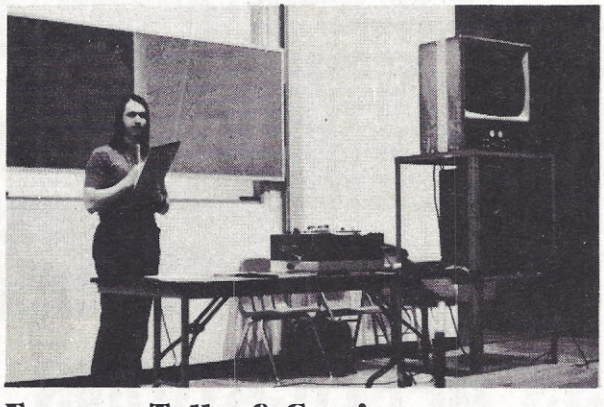
Super Outdoor Flea Market

Surplus computer gear, bargains galore, over 5 acres of space ($5/spot, no electricity).



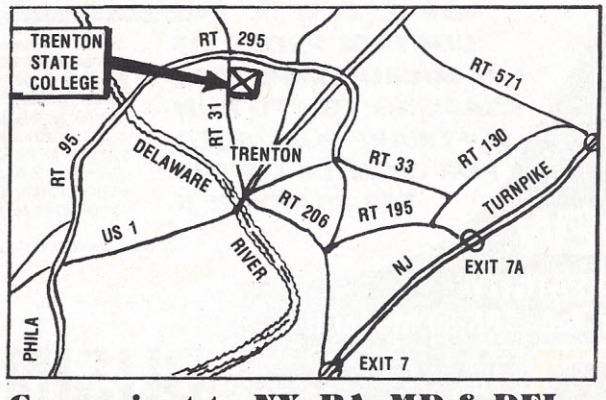
Indoor Commercial Exhibit Area

90 exhibitor booths showing newest products; special discounts; funky games to play.



Forums, Talks & Seminars

Meet the leading experts and hear sessions on robots, computer music, amateur radio, etc.



Convenient to NY, PA, MD & DEL

Easy to get to; free parking for over 6,000 cars.

Free Short Courses on Sunday

Hundreds of Door Prizes • Banquet Saturday Night

For additional information call 609-771-2487

Admission $5 - Students $2

Tickets available only at door

Banquet $10. **Avoid disappointment – purchase tickets early on Saturday at door.**

Sponsored by:

Amateur Computer Group of New Jersey
Philadelphia Area Computer Society
Trenton State College Digital Computer Society
Dept. of Engineering Technology, Trenton State College
I.E.E.E., Princeton Section

and 16 respectively. It was not possible to choose more terms from coefficient file CONWT180 (Figure 11) since we had arbitrarily limited the number to 12 and to have exceeded 8 without going above 12 would have required choosing certain coefficients, while ignoring others of equal magnitude. Figure 16 would probably have looked somewhat better if more terms could have been included.

WHAT ARE THE REAL PERIODS?

If we divide the harmonic numbers (N) of the significant coefficients into the assumed period, we obtain the periods, in days (not counting weekends), of these harmonics. The stock price can be said to be periodic within those periods. Table 2 shows a list of the periods, in days, of the first 20 harmonics of coefficient files CONWT, CONWT 210, and CONWT180. Those marked with an asterisk correspond to significantly large coefficients. It can therefore be tentatively concluded that the stock of Northwest Industries, Inc. during the period from roughly September 1976 to August 1977 fluctuated periodically with periods which are multiples of 12 and 15 days. More computer runs on Program 4YESYNTH is the region of perhaps 175 and 185 days and more detailed analysis would undoubtedly refine these answers. There is also the possibility that the base period may itself be fluctuating, perhaps even periodically.

Table 2.

	Assumed Period 241	210	180
	241 *	210 *	180 *
	120 *	105 *	90 *
	80 *	70 *	60 *
	60 *	53	45 *
	48 *	42 *	36 *
	40 *	35	30 *
Periods	34.4	30 *	25.7*
	30 *	26.3	22.5*
of	26.8	23.3*	20 *
	24 *	21	18
significant	21.9	19.1*	16.4*
	20.1	17.5	15 *
harmonics	18.5*	16.2*	13.8
	17.2	15 *	12.9
	16.1	14	12 *
	15.1*	13.1*	11.3*
	14.2	12.4*	10.6
	13.4*	11.7	10
	12.7*	11.1	9.5*
	12.1	10.5	9

CONCLUSIONS

What do we conclude from these studies?

1. We now have the knowledge and the tools with which to determine the periodicities of a periodic function, determine whether or not an arbitrary function has periodicity, and in either case reconstruct the original function from the Fourier coefficients.
2. Of extreme importance to some people, and ho-hum to others, we have been able to determine that the prices of some stocks, at certain times, fluctuate periodically; and that it is possible to determine, a posteriori, which stocks, when, and with what periods.

The reader should recognize that Fourier analysis is a solid and respected procedure, indulged in by many of the best people. The application of Fourier analysis to stock prices however is a vast and complex procedure, steeped in controversy, argued by many false prophets, and filled with pitfalls for the unwary. Much has been done that is far beyond the scope of these articles, and much remains to be done. The author feels obligated to inform those unfamiliar with the subject that there are other methods used by investigators to solve essentially the same problem. There exists an organization, filled with many brilliant people who share two common interests; investments and microcomputers. For more information contact Jack Williams, Microcomputer Investor's Association, 902 Anderson Dr., Fredericksburg, VA 22401.□

Contact the author at Micro Mathematician, P.O. Box 1234, Cerritos, CA 90701.

CP/M®2 for the TRS-80®Model II

CONTROL PROGRAM FOR MICROCOMPUTERS ENABLING YOU TO RUN SOFTWARE PUBLISHED FOR CP/M 1.4 ON THE TRS-80 MODEL II

CP/M is considered the industry standard disk operating system because it gives you the hardware-independent interface you need to make your computer work for you. CP/M 2.0 is the latest in the evolution of a proven reliable and efficient software system. FMG CORPORATION NOW OFFERS THE CP/M 2.0 FOR THE TRS-80 MODEL II. It features an enhanced upward compatible file system and powerful new random access capabilities. The CP/M 2.0 from FMG provides the ability to run software published for the CP/M system, on the TRS-80 Model II. From minidisks, floppy disks, all the way to high-capacity hard disks, the flexibility of CP/M 2.0 makes it a truly universal operating system. The package includes an 8" system disk, editor, assembler and debugger for the TRS-80 Model II.

CP/M is a registered trademark of Digital Research Corp. TRS-80 is a registered trademark of Radio Shack

VISA master charge

PRICE $200.00

Manual Only $ 25.00

NEW VERSATILITY For Your TRS-80 Model II

FMG CORPORATION

5280 Trail Lake Drive
Suite 13
Ft. Worth, Texas 76133
(817) 294-2510

Call or Write for Complete Information

BUSINESS APPLICATIONS

FROM THE ORIGINATOR OF THE TRS-80 PROJECT

• USER ASSIGNABLE ACCOUNT NUMBERS
• HIGH SPEED ASSEMBLY LANGUAGE PROGRAM
• 18 DIGIT ACCURACY
• AUTOMATIC POSTING TO GENERAL LEDGER
• INVOICE AGING
• CHECK PRINTING WITH INVOICE DETAIL

(Requires minimum 32K, two drives and CP/M)

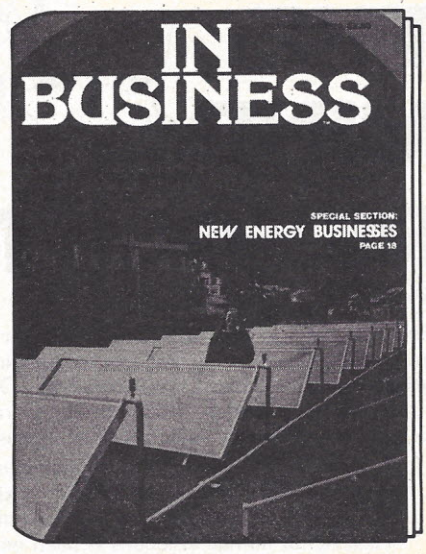
ACCOUNTS RECEIVABLE
Accounts receivable is a low volume invoice system. An entry may be invoiced at any time — before ready for billing, when ready, after billed, even after paid. It even has progress billing which keeps track of milestone payments made at intervals. The program allows automatic posting to the General Ledger and will interface with a future mailing list program for making bulk mailings to customers. Accounts Receivable does not print invoices. Reports:
Not billed
Open and Closed Invoices
Aging Analysis
Customer Statements
Customer Activity List

ACCOUNTS PAYABLE
Accounts Payable is an invoice linked system which means that everything revolves around the invoice. The system provides the user security through the use of a password. It allows automatic (complete or partial) payment of selected invoices, and automatic distribution of each invoice to as many as eleven different general ledger accounts.
This system maintains vendor activity, automatically posts accounts payable and cash accounts, and will interface with a future mailing list program.
Reports: Open and Closed Item Listing
Aging — 30/60/90 days (or user selected)
Transaction printing for Audit Trail
Accounts Payable Ledger

Customization is available at additional cost.

PRICE EACH $250.00

Here is BIG HELP For your small business– or the one you'd like to start!



If you now run your own small business, or if you're dreaming of becoming your own boss and getting paid for doing something that you really enjoy — something that is creative, profitable, and truly meaningful in your life, we've got some good news for you! And that news is: We can help you "make it" on your own.

"We're IN BUSINESS — the first small business magazine that takes your problems, ideals, and values into account.

Emerging New Products, Services and Markets

IN BUSINESS brings you news of income opportunities in a changing world. You'll learn about the best new business ideas in emerging fields of food production and marketing, small restaurants, waste recycling, alternative energy, health care, handcrafts, education, renewable resources, specialty publishing, small-scale farming, community development . . . and many more. You'll discover how to separate the trends from the fads . . . how to tune into today's new consumer . . . how to concentrate on quality and make it pay.

"Like having a top-rate management and financial consultant advise you on a regular basis."

Small Businesses That Are Making It

Every issue will bring you profiles of successful small businesses — how they began, where they obtained capital, how they overcame problems, how they market their goods and services, what their hopes and aspirations are — just about *everything* so you can apply their experiences to your own business.

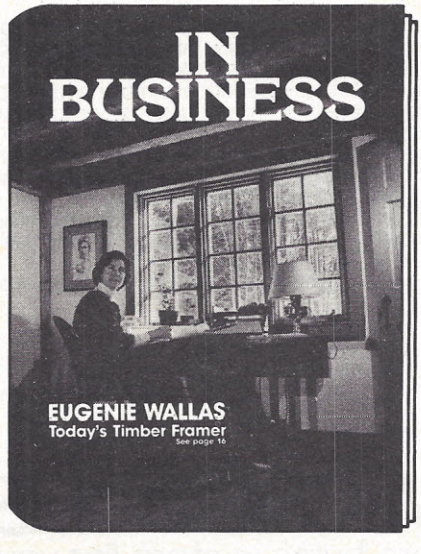
Professional Guidance

Our regular departments and columns will bring you practical guidance for successfully operating a small business — no matter what it may be.

From accounting and advertising techniques to marketing and tax tips . . . from hardware to software . . . whether your business is part-time or full-time, IN BUSINESS will be your source of advice for effectively managing a human-scale enterprise.

GET IN BUSINESS . . . STAY IN BUSINESS . . . WITH THE HELP YOU FIND IN ARTICLES LIKE THESE:

- The Most Successful, Low-Cost Advertising Ideas
- Where to Find Small Business Financing
- Common Legal Pitfalls to Avoid
- Evaluating A New Idea or Product
- How to Breathe Life into a Dying Business
- New Age Enterprise
- Exporting Is Easier Than You Think
- Mail-Order — Making It Work for You
- How to Net $15,000 on 50 Acres
- Tax Credits for the Small Business
- Managing Your Time Profitably
- San Francisco's Solar Center
- Teaching Craftspeople How to Sell
- New England's Tofu Factory
- Building A Customer Prospect List
- Inventory Control Simplified
- Entrepreneuring on Sweat Equity



Mail This Coupon To Try A No-Risk Subscription
IN BUSINESS Magazine
The JG Press, Inc., Box 323, Emmaus, PA 18049

Please send me the latest issue of IN BUSINESS and enter a trial subscription for the term and price I've checked below:

☐ One Year (six issues) $14.00 ☐ Payment Enclosed
☐ Two Years (12 issues) $25.00 ☐ Bill Me Later

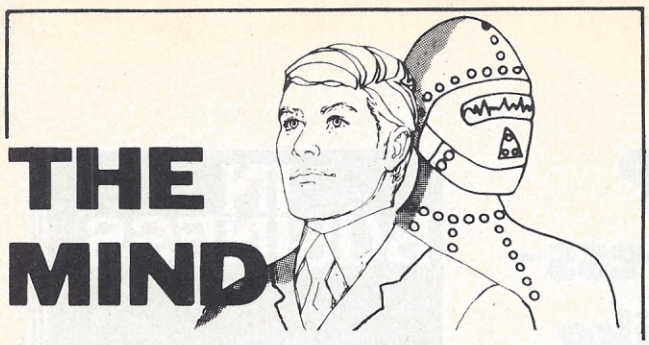
173

Name_____

Address_____

City_____ State_____ Zip_____

(Note: Canadian and overseas subscriptions: add $3 per year.)



THE MIND REVOLUTION

By Merl Miller

The term "robot" was coined in Capek's classic work, *RUR.*[1] Capek had the hero of the play create biological robots that were capable of not only thought, but also emotion. During the time since the play was published in 1929, the term "robot" has come to mean a variety of things. Webster defines it this way:

1. a: a machine that looks like a human being and performs various complex acts (as walking or talking) of a human being; also: a similar but fictional machine whose lack of capacity for human emotions is often emphasized
 b: an efficient, insensitive, often brutalized person
2. An automatic apparatus or device that performs functions ordinarily ascribed to human beings or operates with what appears to be almost human intelligence
3. a mechanism guided by automatic controls

Definition number one probably represents the most commonly held conception of a robot, R2D2 and VINCENT notwithstanding. Many people do not see robots as beneficial; probably because they don't understand what robots are. People tend to expect too much, and ignore the progress that has been made — and a lot of progress has been made in the last few years. People don't get too excited about a simple mechanical arm when they have been exposed to the bionic man. It is no thrill when your human assistant understands what you mean by simple commands, and does what you mean in spite of your clumsy and imprecise language. Similarly, it is no thrill when your machine does the same thing. "Oh, that. . ." is a common response.

As Nels Winkless of the U.S. Robotics Society says, "People are disappointed when the machine doesn't do *better* than a human being at its task in some obvious way. They expect artificial intelligence to be superhuman intelligence — dazzling, mystifying, brilliant. I'm impressed to have a machine with a fraction of the intelligence of the average guinea pig — something smart enough to hide when danger comes, and do something appealing when reward seems available."

Maybe Nels has something there. His machine sounds like the device described in definition three, but then so does an autopilot for an airplane. But, can you convince your computer to respond to the same kind of simple commands you give your hound dog? The answer is no. You can, however, speak to your computer in a clear, precise manner, and get it to follow your directions.

Direct human-to-computer communication has long been a dream of computer users. Right from the beginning, most people realized that the real problem with computers was the input devices. Hence, the reason for voice contact. If we can teach a machine to respond to spoken commands, we should be able to eliminate a lot of errors, and we have taken a big step toward having a talking and listening robot.

The heart of a voice terminal is its acoustic pattern classifier. The classifier produces a digital code when you speak to it. (Your voice produces an *analog* signal that must be converted to something the computer can "hear.") The classifier consists of a spectrum analyzer, and A to D converter (analog to digital), a digital processor, some memory, and an output device.

The spectrum analyzer divides the input speech signal into 16 frequency bands that cover the useful frequency range. This new signal is then fed into the A to D converter where it is changed into digital code. The digital processor converts this code into machine language.

Although my description of a classifier does not include a discussion of such important factors as pattern recognition, pattern classifi-

Our MacroFloppy™ goes twice the distance. For $695.



Introducing the Micropolis MacroFloppy™:1041 and :1042 disk drive subsystems. For the S-100/8080/Z-80 bus. Packing 100% more capacity into a 5¼-inch floppy disk than anyone else. 143K bytes, to be exact. For as little as $695.

The MacroFloppy:1041 comes with the Micropolis Mod I floppy packaged inside a protective enclosure (without power supply). And includes an S-100 controller. Interconnect cable. Micropolis BASIC User's Manual. A diskette containing Micropolis BASIC, and a compatible DOS with assembler and editor. The :1041 is even designed to be used either on your desk top, or to be integrated right into your S-100 chassis.

The MacroFloppy:1042 comes with everything the :1041 has, and more. Such as d.c. regulators, its own line voltage power supply, and, to top it off, a striking cover. Making it look right at home just about anywhere.

Both MacroFloppy systems are fully assembled, tested, burned-in, and tested again. For zero start-up pain, and long term reliability. They're also backed up by our famous Micropolis factory warranty.

And both systems are priced just right. $695 for the MacroFloppy:1041 and $795 for the MacroFloppy:1042.

You really couldn't ask for anything more.

At Micropolis, we have more bytes in store for you.

For a descriptive brochure, in the U.S. call or write Micropolis Corporation, 7959 Deering Avenue, Canoga Park, California 91304. Phone (213) 703-1121.

Or better yet, see your local dealer.

MICROPOLIS™
More bytes in store for you.

cation, and coding compression, it should still give you some idea of how voice recognition works. These are important topics that need to be considered when we discuss the possibility of *just* teaching our robot to listen to what we say.

There are some other refinements to this problem. For instance, we can teach the computer to be a little more polite. One system that does this admirably is Savvy. Here is how it works.

- Suppose a new system in your office will give you weather information which is of interest, because you are a commodity broker. Next to your desk is a terminal with Savvy. You try the system, typing in your commands. You look in the instruction book to puzzle out the command necessary for information you want, then you get the machine's attention.

 "What do you want me to do?" says the machine on its monitor.

 You painfully type in the instruction code: "Ww/76*34& //ss1888**/40axu".

 "OK," says the machine, "what will you say when you want me to do that?"

 "Give me next week's worldwide wheat weather forecast," you say.

 "OK," says the machine, and as the information fades from the screen, you realize you can't remember what your plain language command was. You forget that almost as fast as the complex code. No matter.

 You type: "Show me the weather forecast for worldwide wheat regions." And the machine does it, having figured out what you really meant. In fact, it will respond as well to "Report on next week's wheet waether worldwide," or any other phrase that obviously means the same thing, even if your spelling isn't too good.
- Or suppose you are haggling with the sales manager about the tendency of his customers to not pay their bills, and you need some specific information. With a conventional system, you'd have to ask your computer department to design a special report for you, and schedule a time to run it on the central system. But you have a terminal with Savvy, and you type in: "Can we talk about accounts receivable now?"

 "Yes," says the machine, reaching out for its big fund of knowledge about that subject. "What do you want to know?"

 "Please show me the thirty, sixty, ninety day spread of receivables on customers for our Model 700."

The machine does it without demanding precise, detailed instructions from you. Of course, it has already received those instructions from the computer experts in the back room, but it now interprets your plain language questions about accounts receivable without bothering those people.

Savvy neither aids nor interferes with the ordinary operations of the existing computer systems in these cases. It merely helps the user who doesn't know anything about the format processes of the system, and who can't remember the precise commands, to use plain language in getting the machine's help.

But Savvy *can* be integrated with any system to let the non-expert user do programs in plain language, build files of useful facts for quick personal recall, to communicate with other people and systems, to process words, and to do the thousand tasks a human assistant might perform with the same combination of plain talk and common sense.

Developed by an Albuquerque firm, Savvy is a mix of special hardware and special software; a small version may involve just a single microprocessor board, perhaps 4K bytes of RAM, and twin mini-floppies. Trivial in size and cost, astonishingly capable. Savvy learns the user's own language, and then does what he means when he uses that language.

This common sense approach may be the way of the future, and it may not. Can we design machines that think and feel, or will their limitations be forever set? A robot can be defined as a machine which can be programmed to carry out a series of complex manipulative operations preselected from a wide variety of possibilities. But, is this all there is to it? How will robots serve us in the future? In fact, how will electronics affect everyone's life? These are the kinds of questions I will try to examine in the coming months as the emphasis of this column moves from a study of artificial intelligence to a study of the future.□

REFERENCE

1. *RUR*, Karl Capek, Pocket Books.

Contact the author at P.O. Box 1234, Cerritos, CA 90701.

MetaFloppy™ goes beyond.

The Micropolis MetaFloppy™ gives you more than four times the capacity of anyone else's 5¼-inch floppy. Because it uses 77 tracks instead of the usual 35.

The field-proven MetaFloppy, with thousands of units delivered, comes in a complete family of models. And, like our MacroFloppy™ family of disk drives, MetaFloppy is designed for the S-100/8080/Z-80 bus.

For maximum capacity, choose our new MetaFloppy:1054 system. Which actually provides you with more than a million bytes of reliable on-line storage. For less money than you'd believe possible.

The MetaFloppy:1054 comes complete with four drives in dual configuration. A controller. Power supply. Chassis. Enclosure. All cabling. A new BASIC software package. And a DOS with assembler and editor. There's even a built-in Autoload ROM to eliminate tiresome button pushing.

If that's more storage than you need right now, try our MetaFloppy:1053, with 630,000 bytes on-line. Or our MetaFloppy:1043, with 315,000 bytes on-line. Either way, you can expand to over a million bytes on-line in easy stages, when you need to. Or want to.

In other words, if your application keeps growing, we've got you covered. With MetaFloppy.

The system that goes beyond the floppy.

For a descriptive brochure, in the U.S. call or write Micropolis Corporation, 7959 Deering Avenue, Canoga Park, California 91304. Phone (213) 703-1121.

Or better yet, see your local dealer.



MICROPOLIS™

More bytes in store for you.

INDUSTRY MEMO

By Thomas Carroll
Universal Robot Systems
7025 El Paseo Street, Long Beach, CA 90815

The word 'robot' has different meanings to different segments of society. To the general public and science fiction buffs, a robot is an intelligent machine made in man's image and able to do man's work as he would do it.

The scientists at California Institute of Technology's Jet Propulsion Laboratory in Pasadena envision a robot as an intelligent machine able to crawl across Titan's (moon of Saturn) surface and take measurements and samples. Two-way radio communications with Earth would take hours, so the surface robot must make its own decisions to avoid pitfalls.

The productions engineer sees the robot as a programmable welder for cars in an assembly line, with its long arm snaking in and out making numerous welds. The next day, this same robot can be fitted with a new spray-head 'hand' and be programmed to spray paint the cars it welded the day before. Others of this type actually assemble watches, load and unload hot forgings from giant presses and do other unpleasant and dangerous work.

Ever since the word "Robot" first appeared in Karl Capek's play *RUR* (Rossums Universal Robots), everyone has had their own idea of what a robot is. In Czech, the word is simply derived from the word *Robata,* meaning slave worker. To be formed in man's image is truly inefficient for a machine, but it is these human-appearing robots that capture the imagination.

Because of this recent interest in robots, several firms have been formed to produce 'robot' automatons. Most of these remotely controlled 'robots' are used in promotions and trade shows. Gene Beley of Android Amusement has recently introduced a line of sophisticated remotely controlled robot mannequins that have gained much TV and newspaper publicity. Dave Colman of Colorado has been showing Ralph Roger Robot for many years, and has several smaller robots to contrast Ralph's 8-foot hulking presence. (See article in this issue.) Universal Robot Systems has been primarily involved with consulting and custom design of promotional robot systems for others for several years.

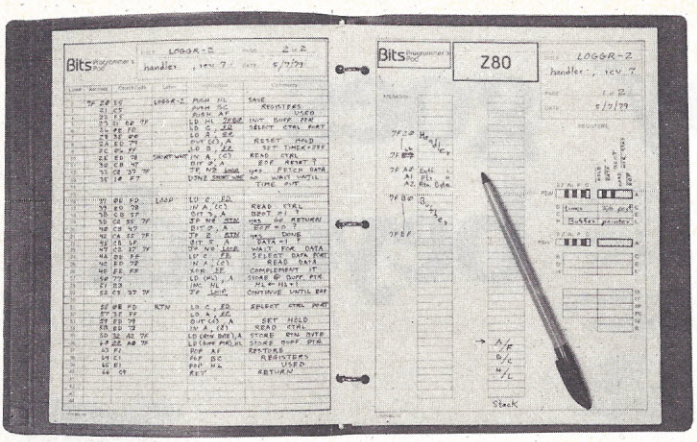
There are several firms in the East that have unique robot lines. Sally Unlimited of Jacksonville, Florida has a series of stationary, programmable mannequins that have highlighted displays from NASA's space exhibits to trade shows. Quasar Industries, Robots Internationale, and Digitec continue to show their radio-controlled robots. Several individuals such as Ben Skora of Chicago have sophisticated robots that tour their local areas.

This is in no way a complete list of all those involved in just promotional robots. Several organizations or publications that may be of interest to the personal computer/robots enthusiast are the United States Robotics Society now based in Palo Alto, California. After a series of reorganizing pains in 1979, the society is now on a firm footing for the robotics enthusiast. Contact John Peers, USRS, 616 University Avenue, Palo Alto, CA 94301. A new publication of noteworthy quality, *Robotics Age,* has started publication to cover the serious robotics hobbyist interests and industrial robotics research. For subscription information contact Phil Flora, Robotics Age, P.O. Box 4029, Houston, TX 77210.

Several new books on robot subjects were published in 1979. The majority were written by hobbyists covering their particular robot projects. Several were well-written, only to be ruined by poor printing and photo reproduction by the publisher. Several others contain outdated material connected only by the authors' rambling thoughts. These books are available at your computer or electronics store, or possibly a technical book store. Judge for yourself.

It might be recommended for the beginner to obtain a copy of Todd Loofbourrow's fine book *How to Build A Computer Controlled Robot,* and augment this with several of the well-written 'Bug Books.' For those experienced individuals with an up and running system who are tired of balancing checkbooks and zapping Klingons, give artificial intelligence and robotics a try. A little imagination, a few motors, parts and interface circuits, and you might open a new world for yourself.□

BITS PROGRAMMER PADS™



For Fastest Service
CALL TOLL FREE 800-258-5477
FOR BANK CARD ORDERS
(in N.H. 924-3355)

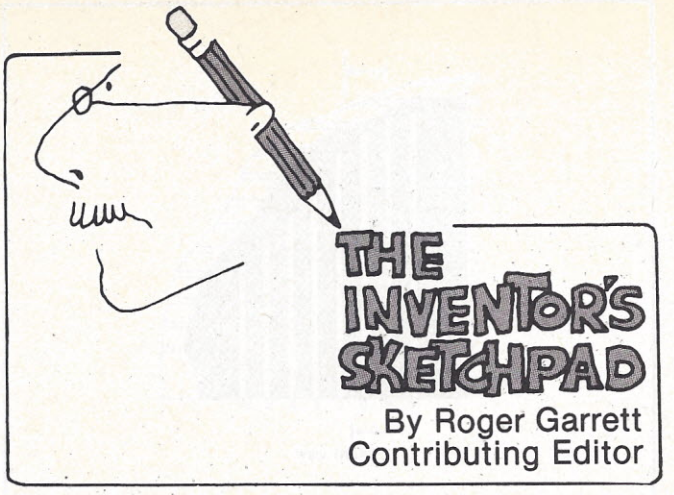
Good programming deserves good documentation. BITS Inc has developed a programming form to help assembly language programmers write and preserve their programs in a loose-leaf notebook format. **BITS PROGRAMMER PADS™** are now available for the **8080A, Z-80, 1802, 6800,** and **6502** microprocessors. On one side of the form the proccessor's register architecture is laid out along with continuous memory locations. This allows the details of your program's register use, stack manipulations, indexed addressing, and table and data storage to be permanently recorded. The other side is for your program or subroutine listing. Its source listing is entered in the instructions, labels and comments columns. Assembling is done next by filling in the object code column which is wide enough for two or three byte instructions. Memory locations are assigned in the address column. If revision or relocation of the program is necessary, the address column can be renewed using typewriter correction tape (a white paper tape about 1/3" wide) and memory addresses reassigned.

BITS PROGRAMMER PADS will protect the effort you have put into your programs and take some of the pain out of hand assembly. Each 50-page pad is printed on durable stock paper, and prepunched for a standard three-ring notebook. They are available for $2.50 each. (Postage and handling: Domestic – 75¢ for one pad, $1.25 for two or more; Foreign – $1.00 per pad to a maximum of $4.00) Please specify which microproccessor.

FREE CATALOG of over 200 Microcomputer Books

BITS inc Books to erase the impossible
POB 428, 25 Route 101 West, Peterborough NH 03458

IA020



By Roger Garrett
Contributing Editor

A PLASMA DISPLAY WITH COLOR

I have been fascinated by the idea of flat panel display screens for some time. You may recall that last year I wrote about a method for producing a solid state panel using a grown-in-place method of LED production. While I mistakenly specified silicon for the crystals instead of gallium-arsenide or other LED-type crystals, the basic method still holds promise for future panel production. I have heard that Motorola will be coming out with a true solid-state display panel and that Sinclair has already developed a prototype of what they call simply a flat-screen television. It will be interesting to see what type of technology they have used to achieve their breakthroughs.

There is another method, however, of producing flat displays; one which is not solid state. It is *plasma panels.* The problem with this type of panel, however, is that it can only produce one color, normally an orangey-red. Certainly that is not acceptable for television type displays. But wait. I have an idea on how they can be made in any color we wish. We'll get to that in a moment. First let me explain the basics of plasma displays.

As in most raster scan displays, a plasma display addresses the screen as a set of individually addressable points or *pixels* (picture elements). The major difference between the raster addressing which broadcast television utilizes and that which plasma panels utilize is the fact that televisions scan (address) the entire set of pixels every thirtieth of a second in a specified order and plasma panels allow any pixel to be addressed at any given time, i.e. random access. It accomplishes this by addressing the individual pixels at the intersection of two metal strips.

In Figure 1 we can see the basic construction of a plasma panel, the lower left-hand corner of which is shown here. In a typical plasma panel there are 1024 by 1024 addressable pixels. The panel is made of two layers of glass. The front glass has a set of vertical metal strips bonded to the back surface. The back glass has a set of horizontal metal strips bonded to the front surface. These sets of metal strips are extremely thin, compared to the spaces between adjacent strips, so that if you held the panel up in front of you you could easily see through it. The two glass panes are sealed at the edges (not shown here) and the space between the two panes is filled with a gas, typically neon. Constant voltage potential is applied across the metal strips; the front set of strips with a positive potential and the back set with a negative potential. The potential across any given intersection (i.e. the intersection of a vertical strip and a horizontal strip) is relatively low, perhaps five volts. This potential is always maintained.

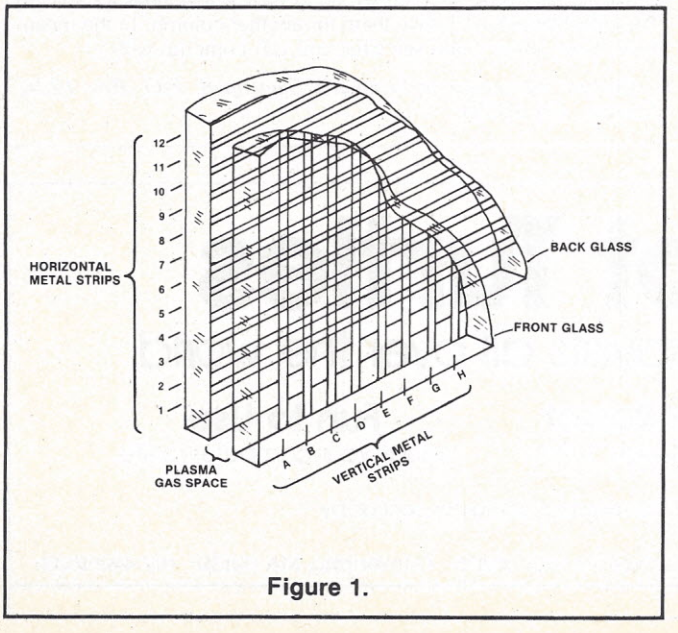


Figure 1.

If only this potential is applied then the entire panel is dark because five volts is insufficient to cause a discharge across the gas space. In order to "turn on" a pixel, a higher voltage is applied across the appropriate intersection. Figure 2 shows a view of our

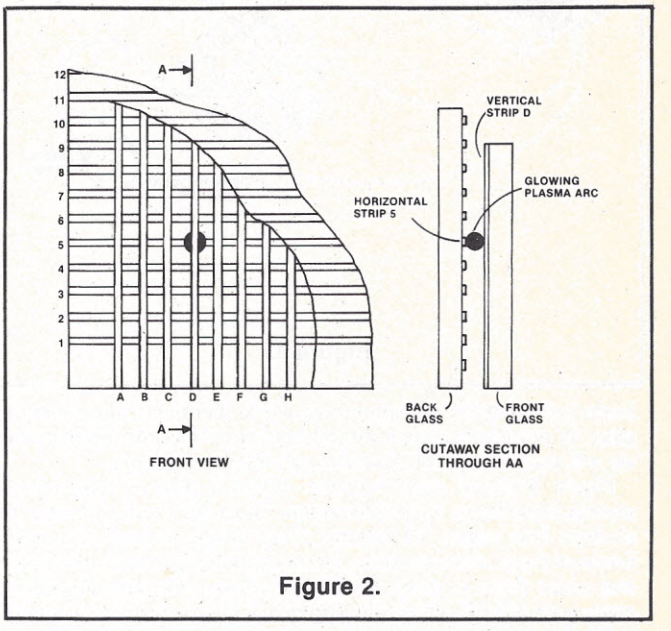


Figure 2.

panel where a high voltage (perhaps ten volts) has been applied at the intersection of vertical line D and horizontal line 5. This is accomplished by applying a positive ten volts to strip D and negative ten volts to strip 5. The only intersection, then, where there is a potential difference of ten volts is at the intersection of D and 5. Ten volts is a strong enough difference to cause a discharge across the intersection. This discharge causes the neon gas to form a plasma which glows, giving off visible light. Since the high potential is very localized at the D5 intersection, the plasma cloud is very small and does not spread throughout the gas space so only that pixel is turned on. Now, if we remove the ten volt potential from the D5 intersection and drop it back down to the normal five volt difference what do you suppose will happen? The pixel will be turned off, right? Wrong. It continues glowing. Five volts is not sufficient to turn a pixel on, it takes ten volts to do that, but once the initial discharge has occurred it only takes five volts to maintain it. So once a pixel is turned on it stays on as long as the lower five volt potential is maintained. It's a sort of write-only memory.

But then how do we turn a pixel off? Simply by lowering the voltage potential across the appropriate intersection to near zero. When it gets near zero volts the pixel goes off. In effect we have a switch at each intersection. We can turn it on by appropriately increasing the potential difference across the intersection and we can turn it off by dropping the difference to below the plasma-maintenance level. We draw pictures or text by simply turning on or off the appropriate pixels.

Now, back to the original problem. The color of the discharge is an orange-red, due to the plasma characteristics of the neon. Yet we may wish to have a green display or a blue display or who-knows-what color display. How can we do it? Well suppose, as shown in Figure 3, we place a fine porous mesh screen between the two glass plates. This mesh is made of some non-conductive material such as fiberglass and is coated with a rare earth phosphour such as the ones used in color televisions. These phosphours glow at particular wavelengths (colors) when excited by electrons. In a TV set an electron

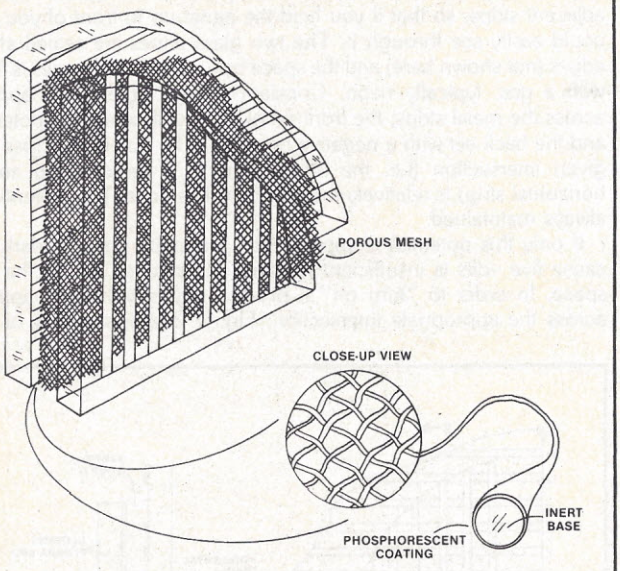


Figure 3.

beam is fired at the phosphours to excite them and cause them to glow. In our plasma panel we will not use an electron beam, rather we will use the electron flow through the pixel plasma.

Figure 4 shows what happens. When a pixel is turned on a plasma cloud forms at the intersection. Since the mesh is porous the electron flow within the plasma cloud travels through the mesh. Some of these electrons strike the phosphours bonded to the mesh and cause them to glow. If we have chosen the right phosphour coating we will get a blue glowing dot. If the mesh is close enough to the front glass panel, the blue color will outshine the rather dim red-orange glow so we will see only the blue. We can now have plasma panels of any color. All we have to do is specify the right phosphour.

Two problems remain, however. First, we now have a red (or green or yellow or whatever) plasma panel display. We still do not have a full color panel. That is, we have an entire panel that is the same color. We cannot, under software control, specify the color of a pixel because the phosphour of the mesh dictates the single available color. So how can we make the panel capable of displaying several different colors at the same time? Secondly, how can we make such a panel into a television-like panel? The answer to the first question is fairly straightforward. The second one is more of a problem because, in order to obtain television quality pictures we must not only be able to display several colors we must also be able

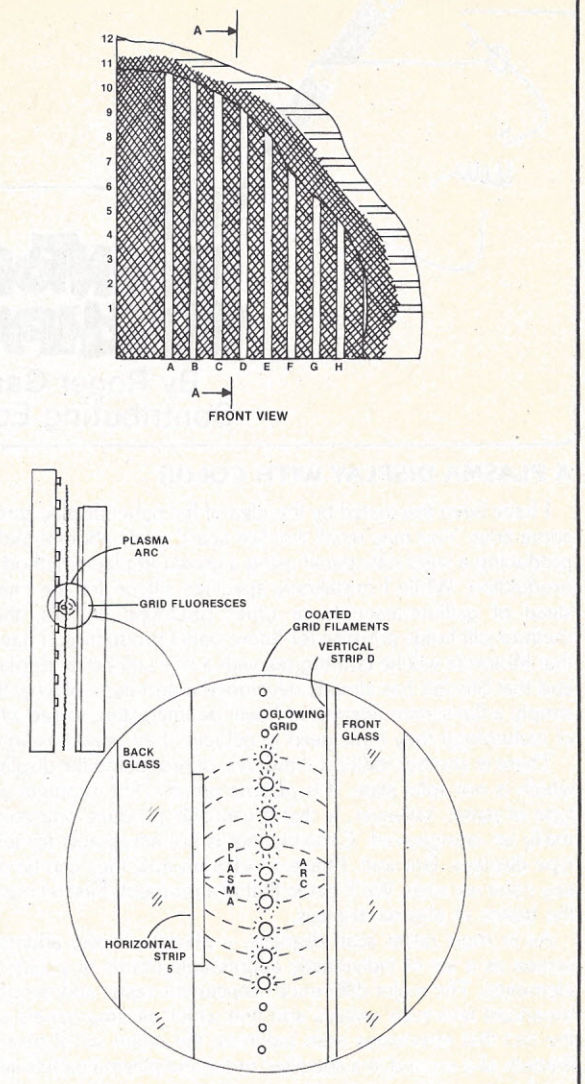
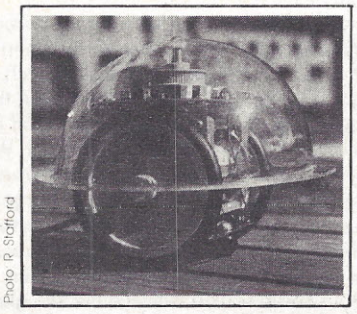


Figure 4.

to display a full range of grey scales. Remember that we can only turn a pixel on or off. It is either fully bright or fully dark. Television requires a full range of intensities and we simply cannot do that with a plasma panel. So what do we do? I have a few ideas for solving these problems but I'll save them for another column. In the meantime, think about it and see what you can come up with.□

Contact the author at Inventor's Sketchpad, P.O. Box 1234, Cerritos, CA 90701.


Photo R Stafford

Robot Turtles

delight hobbyists all over the world

Your Own Robot
senses, draws, beeps, blinks

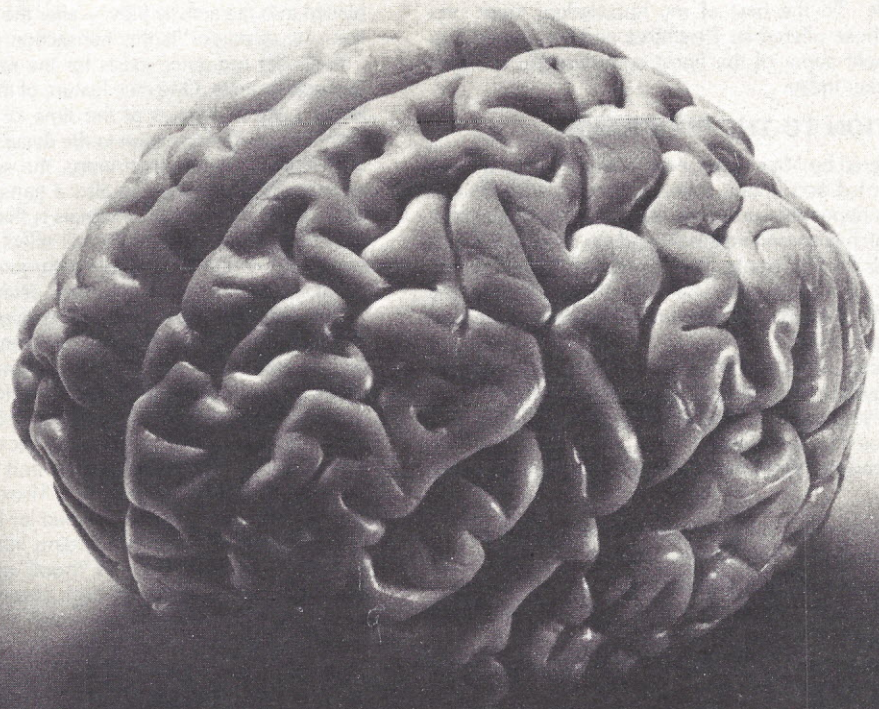
Fun to Use
brings your computer to life

The Terrapin™ Turtle, a robot requiring computer control, by

Terrapin, Inc.

678 Massachusetts Ave, #205 Cambridge, MA 02139 617-482-1033

THE ULTIMATE INFORMATION MANAGEMENT SYSTEM



The brain is the perfect information management system.

Like the brain, we at Micro-Ap specialize in the management of data.

Our software is the state of the art and is designed to efficiently store and instantly report the information you need for your business and personal needs.

From inventory control to mailing list management, Micro-Ap provides the most cost effective software available.

At the heart of our systems are Micro-Ap's unique indexing and reporting methods. You are not limited to single key retrieval. Information can be referenced by zip code, date, name, or any other indices required. Operation is "menu driven" and uses screen displays with all the instructions and error sensing that allow the novice to quickly learn the system and accomplish a multitude of tasks.

A few of the reasons Micro-Ap is so popular are:

- **Quality Software** — It's designed with one goal in mind . . . flawless performance.
- **Customer Responsiveness** — We ask for and receive user critiques and suggestions. All are evaluated and most are implemented.
- **Usability** — We believe that to provide maximum service to our licensees, software distribution should include source code. Ours does.
- **Non-obsolescence** — Our products are continually evolving and our policy is to provide new releases at the current difference in price.
- **Distribution** — World wide by distributors, OEM's, retailers, systems houses, and consultants.
- **Experience** — SELECTOR has been around, and improving, longer than any other data base system in micro-computers.

See the top-rated SELECTOR III-C2, data base manager and the new standard setting GLector, general ledger system at YOUR LOCAL COMPUTER STORE.

If not locally stocked, contact:

Micro-Ap
9807 Davona Dr.
San Ramon, Ca. 94583
(415) 828-6697

Lifeboat
2248 Broadway
NY, NY 10024
(212) 580-0082
telex 220501

MICRO•AP

The Standard In Information Management Systems

CIRCLE INQUIRY NO. 46



This month the Software Review Column will focus upon two fine packages offered by MicroSource Software: Accounts Payable and Accounts Receivable. To the best of my knowledge, these programs, along with those offered by Peachtree and Structured Systems Group, represent some of the finest commercially available software on the market today.

AN INTRODUCTION TO THE PROGRAMS

The programs offered by MicroSource are sold as part of a complete computer oriented accounting system called "Ledger-Plus." The programs are independent modules, all which share a similar operating system which is an enhancement and variation of MDOS. MDOS, which stands for Micropolis Disk Operating System, is designed to function in the environment of Micropolis quad density 5¼ inch diskette. The system I evaluated was designed to operate on the Vector MZ computer.

Each of the modules of Ledger-Plus system is a self-contained sub-program that may be implemented in conjunction with any or all of the other Ledger-Plus modules. Modules may be added to the initial implementation as the need arises.

The accounts receivable program is designed to do the following tasks:

1. Prepare an alphabetical listing of all customer accounts
2. Display and update account information
3. Record sales, payments and adjustments
4. Display, update and selectively purge account transaction history
5. Prepare accounts receivable posting journals with totals for the general journal entries
6. Prepare sales reports with information by salesman or sales outlet
7. Prepare account and credit status verifications
8. Prepare an analysis of those accounts with "no activity"
9. Prepare an accounts receivable trial balance including a balance-due and delinquency aging report
10. Prepare accounts receivable statements for mailing to customers

Basically, the programs are designed to keep track of the monies due the company by its customers and bill them on a regular basis.

The capacity of the system is as follows: with 100 accounts, a maximum of 2925 transactions can be accommodated before the disk is full. With 800 accounts, a total of 825 transactions can be carried on the disk. In between these two extremes, say 500 accounts, a total of 1725 transactions can be accommodated before the disk runs out of room. Basically, the term "transactions" refers to the monthly transactions, i.e. the number of line items for each month. At the end of each month, the system is purged of the monthly data which has accumulated and the disk is free to accept the next month's data.

The manual takes some time to explain the limitations of the system, and in admirable fashion gives two examples of implementation. The data files are designed to have the following kinds of data:

Name	Address	Zip Code
Sort Name	Phone	Credit Code (4 char)

And, of course, the balances. The presence of the other fields makes the system adaptable for other functions, such as collection efforts. The display function is very useful — it allows you to get a picture of what the file looks like.

Recording a Sale

Transaction data entry is very smooth with this system — one begins with the sales amount, then enters the freight amount, sales tax and the total. This is followed by the invoice data and the sales code. The system adds up the sales, freight and sales tax amounts and summarizes them in an internal register. If the invoice amount does not equal this total, then an error message is issued and the entire transaction is voided and one must start from scratch. Some interesting features of the invoice entry sequence include the invoice number, which must be less than 6 characters long (which can be alphanumeric), and the sales code, which provides a means to categorize sales by department, by product, by salesperson or store. The use of the sales code is optional and a sales report can be generated which organizes the data by these codes.

The receivables system gives the operator the option of exiting the routine at any time — through the use of the escape key, and also gives the operator the opportunity to delete a transaction before it is placed into the activity files — after the operator enters each transaction, the question "is this transaction correct" is asked.

A similar sequence exists for the recording of payments, credits and adjustments. One nice feature of the accounts receivable system is the control it gives at the time of posting. When an operator wishes to post information to the detail files which has been accumulated through transaction entry, the system will prepare a posting journal. Note that there is also a transaction entry journal. The difference between the two journals is that a posting journal will reflect only the transactions which will affect the balance of the accounts receivable — the transaction entry journal has everything in it, including set-ups of new accounts, deletions and changes.

The designers of the accounts receivable system had the accountant in mind when generating the summary totals from the posting process — a complete journal entry is generated, a journal entry which contains all of the accounting conventions (DR and CR entries and a summary total of both to show that it does balance). Additionally, a control total for all of the accounts receivable is shown with the previous balance, debits and credits, and an ending balance shown.

As far as documentation, the MicroSource manuals are unparalleled. They are explicit, designed for first-time users, and complete in every respect — even including information about the care and handling of diskettes, how to "boot" up the system and instructions which carry the reader from each phase in the transaction entry, edit, posting and statement preparation. The instructions are clear and they correspond to what actually happens in the computer. Unlike some of the structured systems program manuals, however, there is not very much said about the programs themselves, or the physical or logical layout of the files.

See the Reports

The reports generated by the programs are readable and after you get used to knowing where to look, they are very conveniently laid out. The programmers tried their best to get everything they could onto a standard 8½ by 11 inch paper — a design problem which can be tough to overcome and still maintain readability.

The statements generated by the system use standard Moore business forms. The format and output looks "just like a big system" and the customer is provided with not only an invoice number and brief description, but also an aging of his account balance.

ACCOUNTS PAYABLE

A sister program to accounts receivable, the accounts payable program, represents another module in the "Ledger-Plus" series and, as such, they share many of the same characteristics. The version which I reviewed was in the process of release, and accordingly, the manual was a copy of the printer's proofs. It was still over an inch thick. The documentation, just like the receivables, is explicit, detailed and well organized.

The basic functions which are performed by the accounts payable program include the following:

1. Preparation of an alphabetical listing of all payable accounts
2. A display or update of all account information
3. The recording of invoices, payments and adjustments

4. The display, update and selective purge of all account transaction histories
5. Preparation of an accounts payable posting journal
6. The automatic preparation of checks
7. Preparation of an accounts payable trial balance including an aging report

The capacity of the system is as follows: with 100 vendor accounts, up to 1100 transactions a month can be handled. If the number of accounts increases to 500, only 700 transactions can be processed before the disk will be full. The manual gives the example of the typical small business with 200 payable accounts could process as many as 1000 transactions a month.

In many ways accounts payable is the mirror image of accounts receivable; in the case of the MicroSource programs, this architectural similarity in the programs is evident. The data fields are similar, and the vendor files resemble the accounts receivable customer files in many ways.

Control

One of the features I especially like about the payables program is the inclusion of a set of control worksheets in the package. These worksheets assist in the manual record keeping aspect of the accounts payable and, as an adjunct to the computer processing, help to establish controls over the operations. The worksheets are well laid out, logical and easy to implement. The designers of the system even included a section in the manual about the implementation of the computer and conversion from manual systems.

OUTPUT

A program is only as good as its output. The MicroSource programs have very well organized printouts which are easy to use and interpret. The listings include a trial balance (which has an aging of accounts payable), a transaction journal which contains all data, an alphabetical listing, a purchase journal, a cash requirements schedule, checks, a check register and history listings.

The trial balance which is printed on 11 by 14 (132 column) paper includes vendor number and name, voucher status, voucher number, invoice number and date, due date and aging, plus a discount. The trial balance shows total year to date purchases and displays a "credit code" which the authors of the program intend as a code to describe discounts, such as "2NT" for 2% net thirty days, for example. The trial balance, as with the other documents, contains a date and pagination to insure good identification.

At the end of each listing, there's a summary of the transaction by type, including purchases, partial pay, manual pay, selection for payment, deferment for payment, returned checks, returned merchandise, overcharge or undercharge adjustments, finance charges, additions, changes and deletions of accounts. Batch totals are also given. These "control totals" are very useful as a check on what has been entered.

The alphabetical listing of vendors gives the name, address, phone, credit terms, amount owed and year to date purchases. This particular list would be useful in evaluating volume with various vendors, setting up vendor files or making mailings to vendors.

The cash requirements schedule lists the vendors by vendor number, indicates invoice date and due date, invoice amount and discount amount, with a net of discount amount shown. Alongside each amount is a blank line with the column heading "action required." This list gives an accountant or controller the ability to specify what to pay and when.

The check printing program uses a standard Moore business form check. The top half of the check form serves as a description of what and how much is being paid. One slightly annoying feature of the program occurs when the actual check amount is being printed: the check amount might show as 0006.24, for example. A better display would have been $6.24. In the case of the program's numbers, nothing is there to prevent a bookkeeper placing a 1 in front of the 0006.24 and a check for 10,006.24 being disbursed. This is a minor problem, however, if adequate controls exist. The check writing program does not write out the check amount (i.e. Six and 24/100) — the designers intending for all checks to be protected using a "check protector."

The system has the capacity to not only generate checks, but also to accept manual checks. In the check register, the two classes of checks are identified as "manual" and "auto." The disbursements

NEW FROM RACET COMPUTES DISK SORT MERGE 'DSM' FOR MOD I AND MOD II** TRS-80™

FAST → Now you can sort an 85K diskette in less than 3 minutes* ← FAST

Perfect for your multi-diskette RANDOM file mailing lists, inventory, etc. Ideal for specialized report generation, sort, merge or combination. All machine language stand-alone package — Efficient and easy to use. No separate key files required! Physical records are rearranged on diskette! Supports multiple sub records per sector including optional sector spanning. Sorts on one or more fields — ascending or descending. Sort fields within records may be character, integer, and floating-point binary. Provides optional output field deletion, rearrangement, and padding.

*Sort timings shown below are nominal times. Times will vary based on sort and system configurations. Nominal times based on Mod I 48K 4-drive configuration, 64 byte records, and 5 sort keys.

TYPE	FILE SIZE (Bytes)	SORT TIME (Sec)	TYPE	FILE SIZE (Bytes)	SORT TIME (Sec)
SORT	16K	33	SORT	340K	1081
SORT	32K	49	SORT	680K	2569
SORT	85K	173	SORT and MERGE	85K SORT + 1275K Merge	1757
SORT	170K	445			

DSM for Mod I (Minimum 32K, 2-drives) **$75 On-Disk**
DSM for Mod II (Minimum 64K, 1-drive) **$150 On-Disk****

Mod II Development Package $100**
Machine Language SUPERZAP, plus Editor/Assembler and Disassembler patches.

Mod II Generalized Subroutine Facility 'GSF' $50**

**For Mod II Programs, Include Mod II DOS diskette with order For Development Package, also include copy of Apparat NEWDOS + 5¼ diskette.

∞ **BASIC for Level II and Disk Systems $49.95**
Full MATRIX Functions — 30 BASIC commands!!
Mathematical and common matrix functions. Change arrays in mid-program. Complete array handling. Tape array read and write, including strings. Common subroutine calls.
Over 50 more STRING Functions as BASIC commands!! String manipulation, translation, compression, copying, search, screen control, pointer manipulation and utility functions. Includes multikey multivariable machine language sorts. Load only machine language functions that you want! Where you want in memory! Relocating linking loader! More than you ever expected!!

∞ **BUSINESS (Requires Infinite BASIC) $29.95**
20 Business oriented functions including:
Printer Automatic Pagination with headers and footers!
Packed Decimal Arithmetic (+,-,*,/) 127 digits!
Binary array searches and hash code generator!

COMPROC Command Processor for Disk Systems $19.95
Auto your disk to perform any sequence of DOS commands, machine language loads, BASIC, memory size, run program, respond to input statements, etc. Single BASIC command file defines execution! Includes auto key-debounce, screen print and lower case software driver.

REMODEL + PROLOAD Specify 16, 32, or 48K Memory $34.95
RENumber any portion or all of BASIC program. MOve any portion of program from one location to another. DELete program lines. MERGE all or any portion from tape. Save and verify portion or all of combined merged programs to tape.

GSF (Specify 16, 32, or 48K) $24.95
18 Machine language routines. Includes RACET sorts.

TRS-80 IS A REGISTERED TRADEMARK OF TANDY CORPORATION

CHECK, VISA, M/C, C.O.D.
Calif. Residents add 6%
Telephone Orders Accepted (714) 637-5016

WHEN ORDERING PLEASE ADVISE PUBLICATION SOURCE

RACET COMPUTES
702 Palmdale, Orange CA 92665

register organizes the disbursements according to vendor number, intermixing the manual and auto checks. At the end of the disbursements journal, an accounting entry for the general ledger is summarized and printed.

The history listing gives a complete summary of the activity by vendor on file. The list includes each voucher and applicable payments. While the listing is well laid out, purchases and payments are both shown as positive numbers. A better approach would have shown purchases as positive numbers and all payments and adjustments as negative numbers. The designers of the system instead rely upon the check number printed beside the amount column as an indicator of items representing payments.

Periodically, the paid vouchers and their corresponding checks must be removed from the files to give more room for additional data. The system gives a little "copy and purge report," which outlines the status of number of accounts which have been added and deleted, plus an indication of how many more vouchers may be added to the system before another purge will be necessary.

Now, with all those listings, was there anything missing? At first glance, yes. The purchase journal lists all transactions by vendor, and gives a journal entry by general ledger account number, but does not give a listing of the detail by account number. This type of list would show each account number and voucher number, invoice number, vendor and the description of each item affecting the account. It would provide the detail behind each posting amount to the general ledger account. Such a listing is an auditor's and account analyst's dream. With such a listing, one can easily tell exactly what comprised a debit to a particular account. As it is with the MicroSource programs, one must do some digging to find out what went where.

CONCLUSION

The MicroSource accounts payable and accounts receivable packages represent a significant achievement in excellence in packaged software. They are truly first-time user oriented and, even with their occasional shortcomings, they represent logical choices for the implementation of microcomputer-assisted solutions to real life business problems.□

Contact Carl Heintz at P.O. Box 1234, Cerritos, CA 90701.

MAGSAM T.M.

KEYED FILE MANAGEMENT SYSTEMS
for CBASIC and Micropolis BASIC

Developing business applications in BASIC?

Then you have felt the need to create, retrieve, and update your data by user defined keys. MAGSAM provides this plus much much more...

- Random, sequential, and generic access by key
- Secondary indexing with any number of keys
- Key and record deletes with automatic space reclamation
- Dynamic file allocation and extension
- Complete compatibility with BASIC files
- Easy to use — initiated through simple BASIC statements

In use world wide, MAGSAM is now better than ever!

New! MAGSAM IV™ for CP/M✲ and CBASIC — High performance 8080 assembler version of the highly acclaimed MAGSAM III. Same powerful features with up to 60% faster access time. Ideal for serious business applications where response time is critical **$295/$25†**

New! MAGSAM III™ for Micropolis — Now Micropolis BASIC users can have all the powerful features and capabilities of MAGSAM III. Save development time while providing more sophisticated applications. Perfect for all business applications **$145/$15†**

Improved! MAGSAM III™ for CP/M and CBASIC — MAGSAM III Release 4.0 provides all standard MAGSAM features plus — increased performance; enhanced tutorial program and file utilities; improved data integrity; plus a new memory-saving read-only module **$145/$15†**

Visa and Mastercharge welcome. Dealer and OEM inquiries invited.

† MAGSAM package price/manual only price. Single CPU license. MAGSAM package includes MAGSAM file manager, tutorial program, file dump utility, read-only module (MAGSAM III only), User Guide, Reference Card, and one year update service. CBASIC versions require CBASIC and CP/M-compatible DOS. MAGSAM III distributed as BASIC source. MAGSAM IV distributed in object form. CBASIC versions available on 8" and Micropolis Mod II disks. Micropolis version available on Mod II disk.

✲ Registered trademark of Digital Research

MAG MICRO APPLICATIONS GROUP
7300 CALDUS AVENUE
VAN NUYS, CA 91406

CIRCLE INQUIRY NO. 47

Now—interact with your computer with WILEY SELF-TEACHING GUIDES
Popular, inexpensive paperbacks for homes, offices, and schools

BRAND NEW!

TRS-80 BASIC
Bob Albrecht, Don Inman, & Ramon Zamora
Packed with games, graphics, and practical applications, this eagerly awaited guide leads you step by step to maximum use and enjoyment of your new TRS-80.
336 pp. April 1980 $8.95 paper
"The book is OUTSTANDING....the best thing to happen to LEVEL II owners."—TRS-80 Monthly Newsmagazine

STRUCTURED COBOL
Ruth Ashley
This new guide to the computer language most widely used in business emphasizes structured programming to help make programs easier to understand and check...and less costly to debug. Covers the structure of COBOL, use of unit record files, control logic, tape and disk files, sequential files, random access files, and many other programming techniques.
320 pp. April 1980 $8.95 paper

Over 150,000 copies sold!
BASIC, 2nd Edition
Bob Albrecht, LeRoy Finkel, & Jerald R. Brown
Readers of all ages have used this manual to teach themselves BASIC and apply it to decision making and problem solving in many fields including business, data processing, economics, statistics, education, psychology, and the humanities. This popular guide requires no special math or science background nor even access to a computer.
325 pp. 1978 $7.95 paper
"May possibly be the best bargain in the computer industry."
—Computers in Education

BASIC FOR HOME COMPUTERS
Bob Albrecht, LeRoy Finkel, & Jerald R. Brown
In just days, you can get right down to programming in microsoft BASIC and perform both practical applications and fun and games.
336 pp. 1978 $7.95 paper
"Takes you from knowing almost nothing to knowing almost everything....Albrecht et al. have done it again!"—Kilobaud

INTRODUCTION TO DATA PROCESSING, 2nd Edition
Martin L. Harris in consultation with Nancy B. Stern
Here's a clear introduction to how computers work, what data processing is, how it's organized, types of equipment used, and how a data processing system is designed. This new updated edition incorporates all recent advances in telecommunications, microcomputers, integrated circuits, and intelligent terminals as well as full discussion of computer system uses.
352 pp. 1979 $5.95 paper

Wiley Self-Teaching Guides also teach FORTRAN, Job Control Language, Flowcharting, ATARI BASIC, and Background Math for Computer Problem Solving.

Look for them at your favorite bookstore or computer shop, or write to Pam Byers, Dept. A 7602.

JOHN WILEY & SONS, Inc.
605 Third Avenue
New York, N.Y. 10016
In Canada: 22 Worcester Road, Rexdale, Ontario
Prices subject to change without notice. 0-7602

THE ORIGINAL MAGAZINE FOR OWNERS OF THE TRS-80™* MICROCOMPUTER

SOFTWARE FOR TRS-80™ OWNERS

H&E COMPUTRONICS INC.

MONTHLY NEWSMAGAZINE FOR TRS-80™ OWNERS

MONTHLY NEWSMAGAZINE
Practical Support For Model I & II

- PRACTICAL APPLICATIONS
- BUSINESS
- GAMBLING • GAMES
- EDUCATION
- PERSONAL FINANCE
- BEGINNER'S CORNER
- NEW PRODUCTS
- SOFTWARE EXCHANGE
- MARKET PLACE
- QUESTIONS AND ANSWERS
- PROGRAM PRINTOUTS

. . . . AND MORE

PROGRAMS AND ARTICLES PUBLISHED IN OUR FIRST 12 ISSUES INCLUDE THE FOLLOWING:

- A COMPLETE INCOME TAX PROGRAM (LONG AND SHORT FORM)
- INVENTORY CONTROL
- STOCK MARKET ANALYSIS
- WORD PROCESSING PROGRAM (FOR DISK OR CASSETTE)
- LOWER CASE MODIFICATION FOR YOUR VIDEO MONITOR OR PRINTER
- PAYROLL (FEDERAL TAX WITHHOLDING PROGRAM)
- EXTEND 16-DIGIT ACCURACY TO TRS-80™ FUNCTIONS (SUCH AS SQUARE ROOTS AND TRIGONOMETRIC FUNCTIONS)
- NEW DISK DRIVES FOR YOUR TRS-80™
- PRINTER OPTIONS AVAILABLE FOR YOUR TRS-80™
- A HORSE SELECTION SYSTEM***ARITHMETIC TEACHER
- COMPLETE MAILING LIST PROGRAMS (BOTH FOR DISK OR CASSETTE SEQUENTIAL AND RANDOM ACCESS)
- RANDOM SAMPLING***BAR GRAPH
- CHECKBOOK MAINTENANCE PROGRAM
- LEVEL II UPDATES***LEVEL II INDEX
- CREDIT CARD INFORMATION STORAGE FILE
- BEGINNER'S GUIDE TO MACHINE LANGUAGE AND ASSEMBLY LANGUAGE
- LINE RENUMBERING
- AND CASSETTE TIPS, PROGRAM HINTS, LATEST PRODUCTS COMING SOON (GENERAL LEDGER, ACCOUNTS PAYABLE AND RECEIVABLE, FORTRAN-80, FINANCIAL APPLICATIONS PACKAGE, PROGRAMS FOR HOMEOWNERS, MERGE TWO PROGRAMS, STATISTICAL AND MATHEMATICAL PROGRAMS (BOTH ELEMENTARY AND ADVANCED) . . . AND

FREE

WORD PROCESSING PROGRAM (Cassette or Disk)
For writing letters, text, mailing lists, etc., with each new subscriptions or renewal.

LEVEL II RAM TEST (Cassette or Disk)
Checks random access memory to ensure that all memory locations are working properly

DATA MANAGEMENT SYSTEM (Cassette or Disk)
Complete file management for your * TRS-80™

SEND FOR OUR 36 PAGE SOFTWARE CATALOG (INCLUDING LISTINGS OF HUNDREDS OF TRS-80™ PROGRAMS AVAILABLE ON CASSETTE AND DISKETTE). $2.00 OR **FREE** WITH EACH SUBSCRIPTIONS OR SAMPLE ISSUE.

* TRS-80™ IS A TRADEMARK OF TANDY CORPORATION

H&E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

Box 149 New City, New York 10956

master charge

24 HOUR ORDER LINE
(914) 425-1535

VISA

ONE YEAR SUBSCRIPTION $24
TWO YEAR SUBSCRIPTION $48
SAMPLE OF LATEST ISSUE $ 4
START MY SUBSCRIPTION WITH ISSUE
(#1 - July 1978 • #7 - January 1979 • #12 - June 1979 • #18 - January 1980)
NEW SUBSCRIPTION RENEWAL

CREDIT CARD NUMBER _____ EXP. DATE _____

SIGNATURE _____

NAME _____

ADDRESS _____ STATE _____ ZIP _____

*** ADD $6/YEAR (CANADA, MEXICO) - ADD $12/YEAR AIR MAIL - OUTSIDE OF U.S.A., CANADA & MEXICO ***

Promotional Robots: More Than Mechanical Toys



By Kathy Tekawa, Assistant Editor

Although they are scoffed at by critics in the industry who claim they are mere mechanical toys, promotional robots are becoming creatures in popular demand. With the craze since *Star Wars* and now *Star Trek* it's no wonder promotional robots are superstars, especially in the entertainment industry.

Working on show robots for nearly 16 years, David Colman of Cascade, Colorado is one of the earliest creators.

"People used to give me nicknames because they thought I was crazy spending so much time with my robots. They always told me it would never work. Whenever I'd do shows with robots people would ask, 'what is it?' Now everyone knows what a robot is although it is still not exactly commonplace," says Colman, owner of The Robot Factory, a division of Great Western Display and Production Company.

Colman began his career as a professional ice skater in the Ice Capades and the Ice Follies about 17 years ago. When he was ready to have a new sidekick in his act, he announced he was going to devise an ice skating robot.

It took Colman two years to complete his first robot, using the electronic skills he had learned in the service while working on the Nike Universal, Ajax and Hercules missiles.

Colman's first mechanical ice skater was Commander Robot I, an 8-foot mechanical man who won the hearts of thousands of fans.

After constructing Commander, several smaller creatures were devised, including creeping worms, bugs, and "fuzzies," each having their own amusing personality and functions. But with all the different skating creatures, Commander Robot was still the main attraction of the show.

Commander was constructed with aluminum and plexiglass. The ingenious machine's main mission in life was to ice skate with speed, twirl and perform stunts to keep fans laughing and enjoying the show.

The drive mechanisms inside Commander and his robot relatives who also skate consist of two 24-volt DC gearhead motors supplied by Bodine Electric Co. One motor with right angle drive is placed in each foot. A sprocket on the output shaft drives a chain which, in turn, drives a cog which is held under spring tension against the ice. This causes the robot's 300 pounds to be placed on the ice skates with the drive mechanism counteracting on the ice's uneven surface.

Commander is steered by speeding up one motor or slowing down the other, and it spins when one motor is running forward and the other is in reverse at equal rpm's.

According to Colman, his robots have been a huge success. "When Commander was making his first appearances, the audience would wait just for him to come out, they'd chant, 'We want the robot, we want the robot'."

"The robots did better than the skaters," Colman claims. "The first year Commander missed one show and that's averaging 12-15 shows per week. He had a chain drive broken and we couldn't fix it in time, but I figured the skaters got sick more than that, so Commander had the right to be sick, too."

According to Colman, technology has come a long way from his early robot building days. He recalls his first robots which were controlled with a reed relay system (a relay where two flat magnetic strips are placed inside a coil. When the coil is energized the strips are attracted to each other and the relay contacts are mounted on the strips).

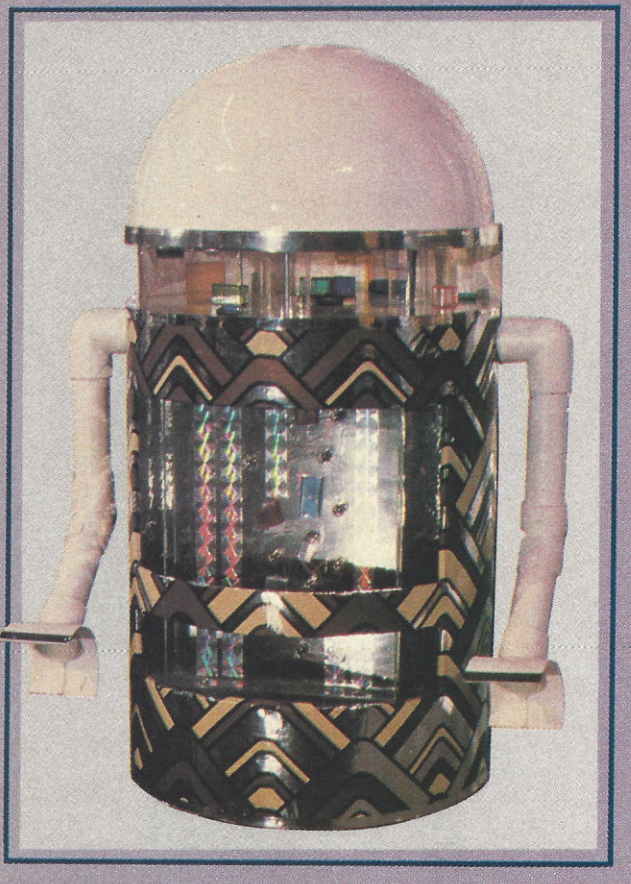
"Now we have remote control which is much more dependable," the robot whiz explains. "Before, everything had to be tuned exactly right in order for anything to work, but now if something is not quite tuned correctly it will still function.

"Also, in the beginning we used mechanical relays and we didn't have variable speed control, but now we use solid-state relays and solid-state speed control," he adds.

Another advancement, according to Colman, which keeps repairs simpler is a chassis containing the parts which slides in and out of a drawer. "We used to assemble everything inside the robot, but now with the chassis we just slide the parts out. It's like the new television sets with the parts in the drawer."

With each robot the creator builds he finds one of the biggest challenges is trying to make them easier to maintain by keeping up with the latest advancements.

Colman's skating robots have worked their way to stardom by performing with the Ice Follies and Holiday on Ice throughout the United States and Canada abroad for the past 15 years.



After Colman left the Ice Follies in September 1978, the Merv Griffin Show wanted to feature one of Colman's robots, Ralph Roger, who is the same type of robot as Commander I. In order for any of Colman's robots to appear on television their feet had to be made to walk on the floor.

After three months of developments and alterations, Colman had Ralph Roger walking instead of skating.

Ralph Roger now shuffles as he bends at the waist and ankles. His arms move up and down from the shoulder and his head moves from side to side; he also talks (using a wireless microphone). His flashing colored lights in his chest can be seen through the transparent framework and his blue eyes either light up or move back and forth, depending upon which model is used.

Since the appearance on the Merv Griffin Show, Ralph is becoming a star, appearing on many programs such as The Johnny Carson, Dinah Show and Steve Allen shows.

Ralph Roger uses the same motors which were in Commander I. "We had to slow the speed down considerably, since Commander ice skated very fast, spun, stood on his toes and bent backwards, but now on the floor he is made to walk slightly faster than a normal pace."

The weight of Ralph is placed on a rubber track similar to that of a bulldozer. The Bodine motors are still used and the robot walks by first running one motor, then the other, and it turns by running one motor forward and the other in reverse. As Ralph walks, the balance is kept by having the weight automatically distributed between both feet.

How does one transport an 8-foot, 300 pound mechanical being? Colman claims, "The packaging is quite tricky with Ralph's size. But the robot does the work by walking into a crate that has a lid which swings down, and at the same time he folds over backwards so that way he is only half the height. Now it's not as difficult to transport."

Both the skating robots and Ralph Roger Robot are operated with a remote control transmitter and receiver with a servomechanism (an automatic feedback control system with some of its signals representing mechanical movement). Microswitches are operated by the servo, which throws the relays in turn.

How much would a machine like Ralph cost? "If you don't count any of the time involved, I would guess the materials cost about $6,000 to $7,000," says Colman who claims it is difficult to be exact on such an indepth project.

After many performances alone, Ralph Roger was joined by another robot Colman devised, an R2-D2 type robot named Six. Six was the sixth robot Colman built, weighing 75 pounds and standing 3 feet, 6 inches tall.

Ralph Roger and Six perform many shows together now, but Six also does solo acts himself. He likes to flirt with women and he is especially fond of children, Colman explains.

For the past three years, Colman estimates he has built 10-12 robots. He hopes to build more mechanical friends and train more operators to run them as he continues his fascinating robot building career.

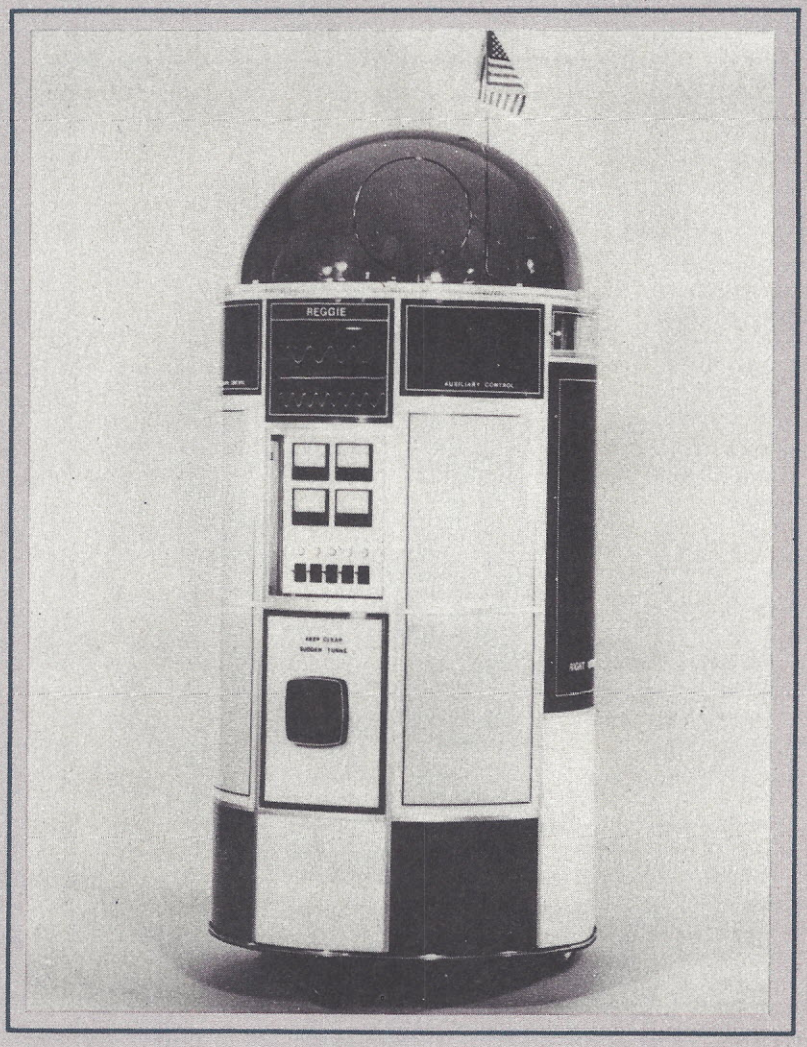
AN INTELLIGENT ENTERTAINER

Robots intrigued him for many years, but it wasn't until three years ago that Ken Davis of General Development Company of Phoenix, Arizona decided to build one himself.

His ingenious creation, Reggie the Robot, which was completed in February of last year, is a moving, red, white and blue computerized machine. Reggie has flashing lights, honking horns, a twirling antenna that moves up and down and when he talks a computerized screen displays his voice.

Davis' intention was not to build a "glorified toy" as he puts it, but a highly technical device. "I thought if I came up with a truly sophisticated, computerized machine there would definitely be a marketplace for it in advertising and promotion," Davis explains.

Reggie, who stands 4 feet, 10 inches tall and weighs 205 pounds, was built primarily for show business. "Although Reggie has self-functions built inside of him, they are not used much because of the show business environment. He needs to interact with people so he only functions by himself with preprogrammed sequences about 30 percent of the time. The operator has to control him about 70 percent," Davis explains.



Reggie's motivation works with two high-torque drive assemblers. He rolls around on two wheels and the third is to keep his balance. Reggie's top speed forward is 2.5 miles per hour and in reverse it's 1.9 miles per hour. "He can push forward 400 pounds and in reverse 300 pounds at full torque, which means whether he is going ½ mile per hour or full speed he can still push all that weight," Davis says.

Reggie is operated with a digital remote control system. He has the capability of performing 512 functions, but is currently programmed for 32, according to his creator.

"All I have to do is give Reggie the direction and he does the rest. With the remote control system I send him the signal and he'll process the information, telling the drive wheels and other electronics what to do," Davis explains.

Although promotional robots are generally known to speak through their operators using a wireless mike, Reggie has three ways he is able to talk in his own voice. "There are preprogrammed sequences inside a digital transporter where his voice is synthesized," Davis says, "or he may have a prerecorded message already inside him or I can use a wireless mike and my voice goes into the machine, is digitized then comes out as his voice. No matter which method is used, the voice is always Reggie's."

Davis and his mechanical companion usually have engagements out of town every four to six weeks. The computerized robot travels in three separate pieces; his dome electronics, his main body and his luggage which carries all his spare parts. Rather than transporting Reggie's 12-volt, 90 amp battery, Davis claims it's easier and cheaper to purchase one locally and discard it after they are finished.

Reggie has eight preprogrammed two minute entertainment routines available. Sometimes he tells about himself, gives lectures, shows off his disco routines or tells jokes while letting out strange squeaks and raising a small American flag. Another unique feature is Reggie's ability to light up inside on command. The audience can then see how he works through his smoked plexiglass.

People's response to Reggie has been overwhelming, Davis claims. "When I was first building Reggie I thought the majority of the audience would be kids, but now I'm finding it's the adults. Although kids are fascinated, adults can really appreciate the technological advancements of the machine. I think they can relate to where it might fit into their lifestyle in the future."

The cost of designing his robot is probably comparable to a small house, Davis says, although he hopes to bring the cost down with the next robots he builds.

Davis, who has several patents in electronics and high technology, is working on Reggie II, III and IV which will be the second generation machines for entertainment also. "It will have an onboard 6800 microprocessor and will be able to function by itself 70 percent of the time."

One major advantage to this is that Davis will be able to train untechnical operators to control the robot; currently he is the only one who operates Reggie.

Davis is also developing a robot for the home with practical functions. "Many people envision mechanical butlers and maids and predict there will be a computer in every home in 10-15 years. I believe this will happen, but there needs to be practical applications made," the mechanical genius says.

To Davis this means the machine should serve the operator rather than the operator serving the machine. "A computer in home or business could be mobile, where by command it could come to you, rather than you going to it. The umbilical cord to the processor is the limitation we still have now. I'm trying to cut that umbilical cord so the operator will no longer be bound to the system."

OVERCOMING THE PERILS AND PITFALLS

The pain and anxiety of prototyping a remote-controlled robot is all too familiar to Gene Beley of Android Amusement Corporation in Arcadia, California.

Beley, who now owns two androids, had to overcome many obstacles before experiencing the rewards of his labor. Beley's interest in robots developed after he worked for Quasar Industries for a short time.

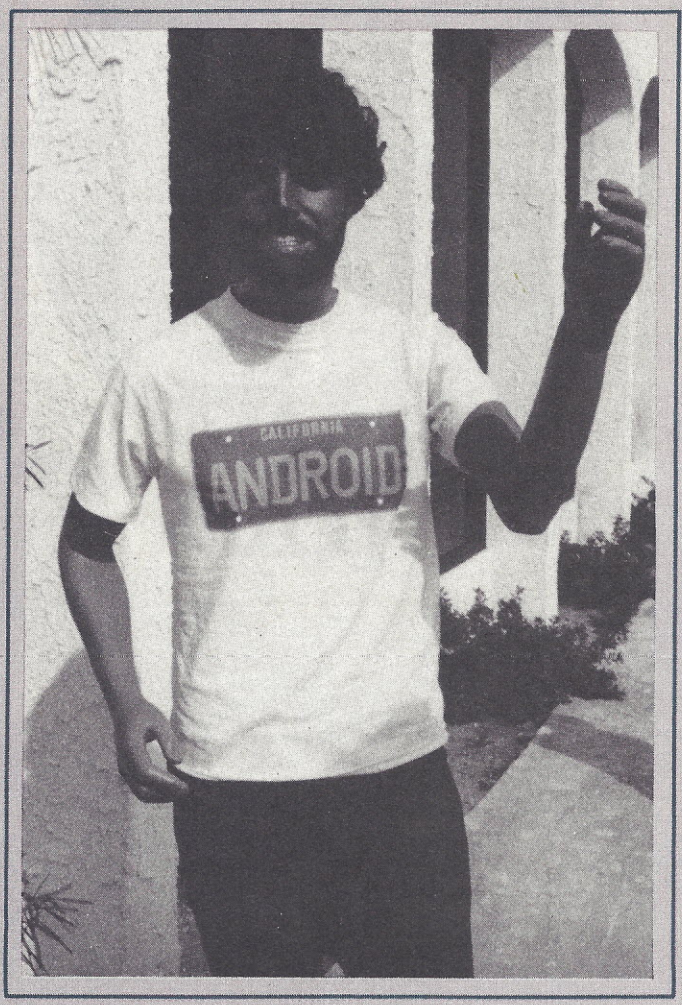
He started his own company (then called Games People Play) and he promoted electronic games and pinball machines. His imagination was triggered even more after he saw *Star Wars* in 1977.

Since Beley had no technical background he looked for a partner to help him develop a robot and share the financial burden. "It was so discouraging, I spent about three months looking for venture capital. Everywhere I went I'd have to spend an hour explaining what a robot was," he says.

When Beley finally met someone who also wanted to build a robot they formed a partnership. Beley's partner, a technician, built a talking robot called AG-1 or Argon (featured in the April, 1979, issue of INTERFACE AGE). The 300 pound, four foot Argon had a color TV built into his chest with six computer games programmed into the unit.

Before Beley and his partner went separate ways he says they experienced one incident all show robot owners must encounter at one time or another.

"At a Las Vegas show with Argon, a man came up to talk to me and all of a sudden there was all this smoke. Sure enough, it was coming up out of Argon's head," Beley laughs as he explains. "We hustled that thing off the floor and took our client to lunch assuring him we would fix it, although we were sweating blood. When we came back, fortunately Argon was fixed and he was back on the floor again."



Beley has learned that building a simple, reliable robot is one of the most important criteria for his business. "If your client is going to pay at least $500 a day and that robot breaks down you better be able to fix it awfully fast, otherwise you aren't going to be in business for long."

He says he has been through all kinds of "crazy events" as he puts it and feels that other robot lovers could gain from the mistakes he has already suffered. Beley is currently writing a book on his stories behind-the-scenes.

The most recent developments of Android Amusement Corporation are two androids, Andrea and Adam. They are human-type mannequins which run by remote-control on a "RobotCart." The production of the two androids took a lot more research and time than Beley ever imagined.

"I never knew what it would take to put together my concepts, and I don't think anyone really knows till they've tried it themselves," he explains. "In phone calls alone I spent hundreds of dollars."

Beley estimates he's spent about $15,000 on his androids although he says there are many expenses unaccounted for. "Usually when there's prototyping about 25 percent are mistakes, such as motors which go unused or other equipment, you never know what will work best until you actually use it," he says.

The basic mannequin alone cost $500 and for custom alterations it was another $500. Why did Beley want human-form robots? According to the entrepreneur, the advantages lie in the ability to change their costumes and wigs, giving them a variety of appearances. Other important advantages of radio operated robots, the android producer points out, are their portability and the ease of operation even by non-technical persons.

"They've made a big hit with the media because they are so human-like and photogenic," Beley says. The robots have appeared on local TV stations and are tentatively scheduled to appear on NBC's TV show, "Real People." Another source which has helped Beley is a book entitled *Only in L.A.*, a list of unusual services that includes the androids.

The RobotCart which the humanoids roll around on is powered by two 12-volt Gel-Cell batteries. There is also a built-in battery charger to recharge the unit and the radio receiver. Inside the unit is a stereo tape deck which is remotely controlled so Andrea and Adam can move to disco beats.

Although the androids move stiffly rather than bending at the elbow or waist, Beley claims they are still very effective. "Of course, there is always room for improvement but the cost that is involved makes it difficult to make improvements all at once," he says.

Beley claims it would be simple for technicians to add to the unit. "Hobbyists could easily piggyback a microprocessor system on to the cart and it could really add a lot. There's room for one to be stored, so it's a definite possibility," he says.

How is the public accepting these humanoid creatures?

According to Beley, "We're past the stage of wondering if they'll be acceptable or not. We already know that our clients are ecstatic over them because it's such an attention getter and really draws in crowds. I think in this day and age of aliens, people enjoy a shock and when you see something that looks human that walks and talks it is quite shocking."

Beley's primary function is not to promote robots that wash dishes or clean the house, but that can help people have fun. "We can never claim to be more than amusement beings. That's why I like to have *amusement* in our company name so we will never be misconstrued," he explains.

Beley hopes that Andrea and Adam will some day be a mass product. He uses the automobile industry as an analogy of his business. "The Model-T wasn't an assembly-line item until Ford came along and developed it as a mass product. That started the whole automobile industry. It's the same thing with robots, it's a matter of breaking that price barrier and seeing who's robot will be the first to make it. I'm waiting for that time to come and we'll just have to be ready."□

AN INEXPENSIVE ROBOT GUIDANCE SYSTEM

"EYES" FOR YOUR ANDROID



By Dave L. Morris

One of man's basic fears is that of being lost. The thought of being blind, deaf, *and* lost is too frightening to even think about. And yet most robot builders apathetically allow their creations to exist in this condition. I have recently become more and more aware of a need for a good obstacle avoidance and navigation system. Some systems developed for other industries have been adapted to robotics with varying degrees of success.

My first attempt at this was in high school, where I devised a computerized student transportation system. Unfortunately, I did not have the $2 million to construct it. Now that my interests have turned to robots, I have come up with some inexpensive ideas to help my current android project get where he is going without my having to hold his manipulator. The fundamentals can be implemented in an inexpensive system with a processor, about 8K bytes of RAM, some cheap electronic components, and some software tricks which I will explain.

The concepts are presented in such a way that a phased, modular approach may be taken to implement the navigation system, beginning with the simplest, and working up to the most complex maneuvers. For those whose level of expertise already includes Reverse Polish Doppler Laser Triangulation, Gastrointestinal Sonar Phase Comparators and the like, you may at any time execute a "JNI END" (jump if not interested), for now I am covering some very necessary basics.

The main components of a robot are illustrated in Figure 1. These are the Main Decision Maker, the Communications Subsystem, the Power Subsystem, the Manipulator Subsystem, and the Transportation Subsystem. It is this latter area that I will be discussing. We are talking about robots of the "android" variety; that is, they look like humans, not the automatic spot welder robots at General Motors which are bolted to the floor.

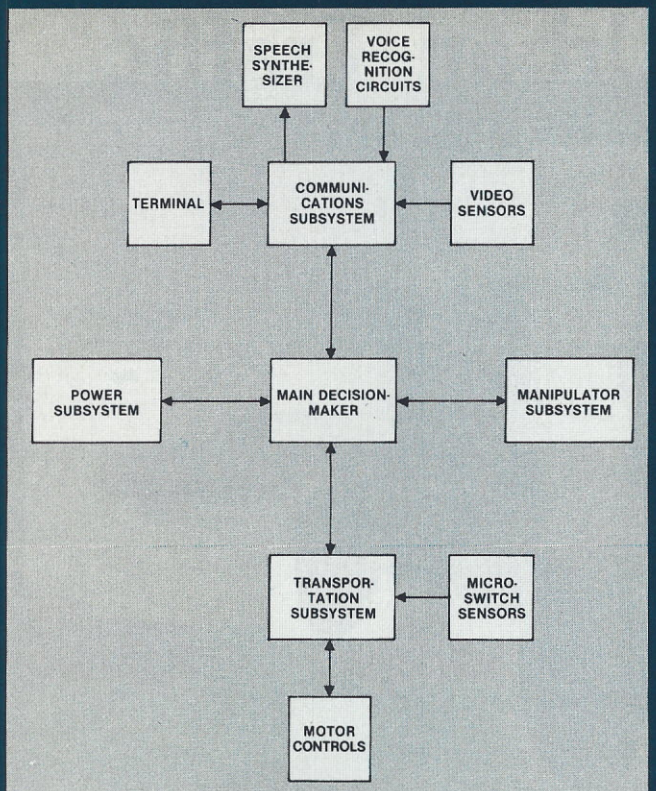


Figure 1. Overall diagram of robot subsystems.

The object then, for the Transportation subsystem, is to get the robot where he is going; using the most direct route, "learning" from past collisions with tables, glass doors, and an occasional unprotected railroad crossing. This means the robot is self-contained and uses no wall-mounted homing beacons or underground wire grid! This is what I call the Pinocchio effect ('I have no strings to hold me up. . .'). That seems to be the simplest way, meaning the cheapest, also.

Let's say we want our robot to be able to get from point A to point B in a maze. A clever way of finding the way through it is to always keep to the right hand wall, following every blind alley if necessary, until you finally get to the destination. We could teach our robot to always stick to the right hand wall. But applying that principle to a robot would be like traveling from Florida to Oregon by taking all right hand turns until you finally get there.

This is not the kind of intelligence desirable in an otherwise "useful" robot. Neither do I need a 200 pound full-grown wind-up toy thrashing around aimlessly. The minimum acceptable system would be one that:

1. Remembers past mistakes (where the obstacles are)
2. Plans trips before driving off (finds a no-obstacle path)
3. Periodically checks to see if obstacles have moved

The path chosen by the robot should be the shortest one, and, unless something has been moved since his initial venture, he should not have to touch the obstacles again.

The transportation/obstacle avoidance system consists of hardware and software. First, we will look at some hardware ideas. The base usually consists of a circular or square base-plate of wood or aluminum which houses the battery, power system, computer and electronics, and drive motors. With a swivel wheel at the rear, a triangular support pattern is established which provides a stable, mobile platform for most any type of superstructure.

For the computer system, a single-board microcomputer, such as the S.D. Systems "Z-80 Starter Kit" would be ideal, since most everything can be contained on one or two boards. This computer costs around $245 and gives you the power of the Z-80. Microswitch "bumpers," mounted on front, rear, and sides tie into the interrupt structure and provide tactile feedback which will be described later.

This, then, comprises our transportation and navigation hardware. Radar, ultrasonic, laser and other such detectors were discarded from the initial project due to cost and complexity. After all, every home should be able to afford a robot, right?

What we have is essentially a blind, deaf, and mute robot that "touches" his way around the rooms with the aid of some strategically placed microswitches on the base-plate. Picture yourself in an acoustically "dead" room, blindfolded, with your hands tied behind your back. If your mission were to proceed to a certain point in a room you had never seen before, you would have to find the table, couch, chairs, bookcases, and other furniture by bumping into them. And so with the robot. But, our system will give him several advantages over you:

1. He will memorize precisely the location of everything he touches;
2. He will know exactly which direction he is facing;
3. He will know exactly where he is in relation to the obstacles and his destination;
4. He could give up and return to his starting point *without touching another obstacle.*

All this can be accomplished in the following way: The robot's "world" is a 2-dimensional array in which obstacles of all kinds are recorded. He can "see" them any time he desires by testing that location in memory.

Each bit in memory has two 8-bit coordinates addressing it. (The "Y" coordinate's 3 low order bits are shifted out and used to address bits 0 through 7 of a byte. That byte is found when the newly shifted "Y" address is concatenated to the "X" byte, so that the most significant 3 bits [MSB] of X are zeroes, and the least significant bits [LSB] have been shifted into the high order "Y" bits. The resulting 16-bit address determines which byte contains the final obstacle bit.

Thus, for example, when X= 0011 0111 and Y=0001 1100. Then, stripping out the 3 LSB, Y= 0000 0011. Then, shifting X into Y, the final result is X=000 0110 and Y=1110 and 0011, or an address of 06E3 [HEX]. The bit addressed is bit 4.) Thus the maxmum size of the array is about 8K bytes. When an obstacle is found, the bit at that location in the "world" is set (binary 1). Every location that is obstacle-free is reset (binary 0).

If the grid has 10 cm increments, then a 50 x 50 foot "world" would occupy less than 3K bytes of RAM. Using a Z-80 processor, the BIT SET, RESET, and TEST instructions make this representation very efficient. Added accuracy could be achieved by reducing the distance between coordinates. However, this means the robot must map out a much more detailed area, or there will be gaps in the map. This would, of course, also require more RAM for the same size world. Figure 2 illustrates the difference between 10 cm increments and 5 cm increments when plotting a typical obstacle, a table.

Increasing the RAM area used for the map will increase the size of the robot's "world." If the robot is to be used indoors only, then the map area need be no larger than the size of the house; otherwise, the map can be either re-drawn every time the boundaries are crossed, or the robot can be restricted to his mapped-out territory. The formula for determining memory requirements for the array is:

$$M = ((L/i)^2)/8$$

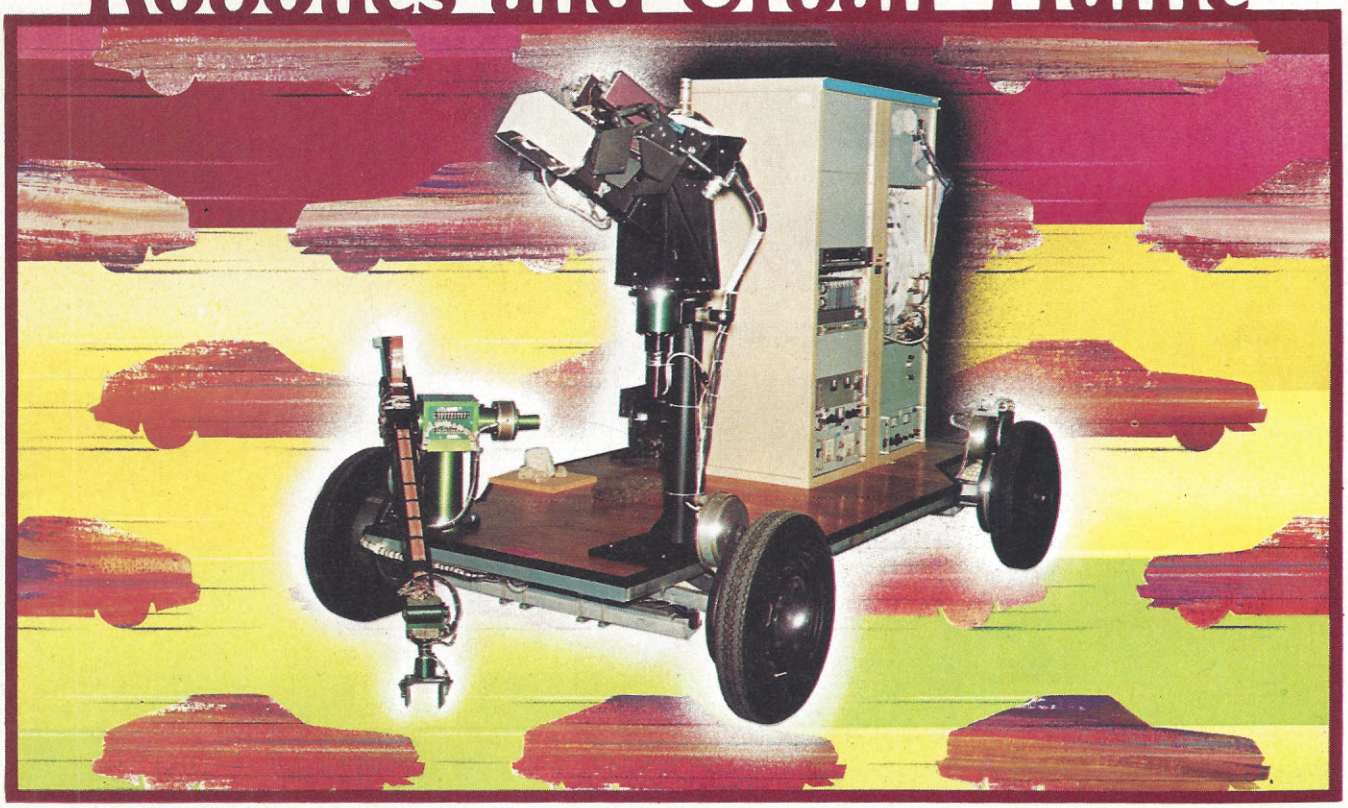
where: M is the required memory in bytes (round up!)
L is the length of one side of the "world".
i is the distance between grid lines

The formula will work for either metric or that other old system of measurement, as long as you stay consistent with the units all around.

The computer must keep track of its position by either "Open-loop" or "Closed-loop" tracking. "Open-loop"

Continued on Page 138

Robotics and Urban Traffic



By Boris Dobrotin
Jet Propulsion Laboratory
California Institute of Technology

Robotics has progressed rapidly with the development of smaller, faster, and more capable computers, and is to the point where computing elements and integrated sensors can monitor an environmental state and make judgmental decisions in real time. Thus, it is timely to examine the problems associated with applying this new automation capability to city traffic.

This process is already underway in many areas. The United States and other countries are developing techniques for automatically guiding unmanned vehicles for use in hazardous environments, while Japan and England are investigating highway guidance techniques.

However, urban traffic automation requires both guidance and routing of vehicular traffic in such a way as to retain the one major feature of modern urban traffic: flexibility. It is this aspect of a personal vehicle which must be retained in order to obtain user acceptance of a fully automated vehicle. The choice of time, destination, and even route are all attractive features of the individual vehicles which make up modern urban traffic. Additionally, this freedom of movement is apparent in the interaction of vehicles within the traffic flow, i.e., the ability to change lanes, enter and exit the traffic flow, and even the spacing of cars relative to one another.

With the aim of maintaining flexibility, autonomous traffic control may be divided into two functions: vehicle routing which selects a path from a trip's beginning to end, and vehicle guidance which ensures that the vehicle follows the desired path in a safe manner. Such a division allows a hierarchical structure to be imposed: a central routine capability, local intersection control, and an on-board individual vehicle guidance system. This allocates information processing capability in the optimum manner. The large off-line computers needed to solve the stochastic routing problem are centralized, the intersection merging control is performed by an on-site minicomputer, and an on-board microprocessor is used for real time vehicle guidance. Intercomputer communications are also minimized, with sensor data being processed in situ, with only minimal information being transferred to the next hierarchical level.

The development status of the routing and guidance problems are not equal. The routing problem has many applications (communications, warehousing, railroads, etc.), and an extensive body of material is available which is directly applicable to traffic control. However, the same cannot be said of vehicle guidance. Many of the studies done on automatic vehicle guidance are aimed at maintaining control over a precise, fixed path; a technique which produces the equivalent of an electronic railroad. For example, wire guidance requires that a guide wire be installed with separate signal frequencies for each lane. This technique requires discrete lane changes including exit and entry points. Furthermore, wire guidance has no inherent provision for obstacle detection and avoidance, requiring ancillary sensors.

What is needed, then, is a vehicle guidance system which allows path selection flexibility, does not require a high degree of precision, and can sense and respond to unexpected events. Such a technique would offer the additional benefit of allowing easy integration with a dynamic routing capability. A solution to this problem exists, and is based on human vehicle control.

A human does not drive precisely, but rather drives relative to external references, such as distance to nearest car (both to one side as well as ahead), as well as maintaining an average path around corners, for example. Thus, a large part of introducing robotics into vehicle control is to provide a real time system capable of providing the human's coordinated vision/decision process to maintain the correct vehicle heading (guidance) and to select the correct route and avoid obstacles (path selection).

What is the prognosis for such a development? Good. For example, at the Jet Propulsion Laboratory a robotics program is in progress, developing these coordinated capabilities

for a variety of space related tasks ranging from surface exploration of Mars to the assembly of large structures in space. A hierarchical computing system has been assembled which uses an off-line computer (DEC PDP-10) to provide routing information, while a local microcomputer (General Automation SPC 16/85) provides local control. Sensor data is processed by a microprocessor (DEC PD 11-03).

The primary related effort in these various applications is the integration of real time vision sensing into a closed loop control with off-line planning. A major element is the capability to obtain and process video data at a high enough rate to guide a vehicle across unknown Mars terrain, or to join two beams in space.

Once the ability is provided to visually sense the vehicle's local environment and its position relative to that environment, the two problems of guidance and path selection can be solved. Unlike other vision-related tasks such as part assembly, vehicle guidance in traffic does not require precision, but operates using relative position knowledge on the order of 2-3 feet (lateral) and 10-20 feet (longitudinal). Also, identification of obstacles does not require detailed knowledge, but basically operates on size and relative motion information.

Vehicle guidance is a dynamic servo control problem. As indicated above, the requirements are not severe, but fast response and stability are needed. For normal operation, low accelerations (approximately 0.1 g) are used, but faster response is needed for emergency situations. The capability of vision to look ahead supplies the required servo system the lead time necessary to insure stability and fast response.

On the other hand, path selection is a kinematics problem with both obstacle identification and relative motion and position data needed so that the preplanned route to the desired destination can be modified to avoid obstacles (i.e., other vehicles) as well as identify intersections, and so forth.

Both of these capabilities have been demonstrated at JPL. A laboratory model of a Mars roving vehicle has been built and tested.[1] The vehicle uses two TV cameras connected to a minicomputer to identify obstacles in the desired path and steer around them. A second vehicle was developed, as a target, for high speed remote control operation. This program served to identify requirements for the remote sensing needed for automotive guidance.[2]

The final key to providing these capabilities in the real world is in the development of simple high speed vision data extraction techniques, also under development at JPL.[3] An "artificial retina" has been developed using a minicomputer and TTL logic in conjunction with a commercial solid state TV camera, and plans are underway to implement this sensor in VLSI under microprocessor control. As an analog to the human eye, pipeline processing provides object boundaries at real time (TV) rates. The combination of high speed information extraction combined with simple obstacle location and motion detection algorithms in an on-vehicle computer are now under development.

Using the basic technology that exists in robotics today, a fully automated urban traffic system is feasible. Integrating onboard vision based guidance with centralized routing schemes offers the capability of retaining the flexibility and individual service of the existing system and the increased safety, throughput and convenience of an automated system.□

REFERENCES

1. "Fully Automated Urban Traffic System," U.S. Department of Transportation Report DOT-TST-78-3, December 1977.
2. Dobrotin, B., "Automated Guidance for Ground Vehicles," Association for Unmanned Vehicle Systems, June 1979.
3. Eskenazi, R., "Low Level Processing for Real Time Image Analysis," JPL Publication 79-79, September 1979.

Teratek's

POSTMASTER

Tomorrow's mail system. Today.

One package does it ALL.
Postmaster offers the most powerful and flexible mail-management system available.

Batch Entry: Entering names and addresses to a mailing list is simple. Repeated elements of a record need only be entered once.

Powerful Record Extraction: Used in conjunction with the **Optional Reference Field,** this feature allows simple creation of user specified "target-files."

Dedicated Record Editor: List, modify or delete records. Allows intact or extracted backup of original file.

Automatic "ID" Field Insertion: (optional) Key in a name, and a unique 10 character record identifier will be entered automatically to the Reference area.

Envelopes: Postmaster prepares single or continuous envelopes.

Mailing Labels: Standard or user-specified formats up to five across are supported by Postmaster. User may specify any number of labels per name.

Form Letters: Prepare and edit form letters in a variety of formats, on either single or continuous forms. Optional capability of allowing text or salutation "Inserts" for some or all letters in any print run.

Dedicated Record Sorting: Sorted files are re-written to disk. The sort may be in either ascending or descending order. Uses the FAST Shell-Metzner sorting algorithm.

Attractive Reports: Neat, paginated reports on either 80 or 132 column paper. The 80 column option allows your CRT to provide an attractive report display.

Clear, Complete Documentation: The manual will explain in simple English how to get started right away. Sample data and form-letter files are included on the disk to allow new users to experiment (learn) quickly.

Quality That's Affordable and Available: The Postmaster programs are available in a variety of 5" and 8" disk formats (40k of RAM, CP/M and CBASIC2 are required). Among the formats supported are TRS-80, North Star, Heath H8 and H89, standard 8" IBM, Vector MZ and other CP/M derivatives capable of running CBASIC.

It's terrific!
We use it ourselves!

Prices reflect distribution on 8" single density diskettes. If a format is requested which requires additional diskettes, a surcharge of $8. per additional diskette will be added.

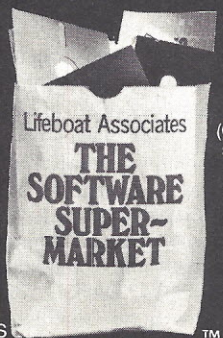
COMPLETE PACKAGE: $150.

MANUAL ALONE: $15.

(Credited toward subsequent purchase)

*CP/M is a trademark of Digital Research

Inquire about additional features!



LIFEBOAT ASSOCIATES™
2248 Broadway, New York, N.Y. 10024 □ (212) 580-0082 □ Telex: 220501

Computer Simulation of Classical and Instrumental Conditioning

By Alan Filsinger
University of Pittsburgh

Traditionally, artificial intelligence (AI) has dealt with such topics as problem solving (e.g. chess programs), pattern recognition, and natural language ("speaking English to your computer"). These are difficult tasks and the programs involved tend to be complex and large. Those who have used programs like "ELIZA" also realize that many natural language programs are based on tricks, and that they do not begin to satisfy Turing's famous definition of machine intelligence.

Another approach to AI is to consider how learning takes place in animals, and to then write a program to simulate this learning. Once elementary animal learning can be simulated, it might be possible to attack situations involving more complex animal learning. Whether the results of such investigation will have any bearing on human learning is open to debate.

While the precise relation between "intelligence" and learning is still largely unknown, many activities classified as intelligent can be traced to learned procedures. Even such an activity as theorem proving is based upon prior learning. Most math students who prove theorems have learned to do so by following learned procedures. They apply what has been learned in a suitable way (also learned) according to learned rules. This sort of activity can be simulated on a computer.

LEARNING IN ANIMALS

There are two fundamental ways in which learning has been observed to occur in animals. These are termed classical and instrumental conditioning. By far, the majority of animal learning can be explained in terms of these two procedures. Many psychologists will argue that at least a part of human learning also involves these processes.[1] Others (e.g. Maslow) find this view too simplistic. The reader who works with the simulation of these processes will undoubtedly develop his or her own opinions about this.

This article will explain these two forms of learning as well as discuss simulating them on a computer. Runs of the program will be presented from a terminal session that simulates the learning process as it takes place. I/O will be presented for each type of learning. Finally, a combined terminal session will be shown illustrating both processes taking place and how they interrelate. A listing of the program which performs the simulation is presented in an appendix together with an explanation of the model on which it is based.

The relatively small amounts of memory required by this program make it possible to do such simulation via microcomputers. Thus the possibility arises for this form of AI research to be done on relatively small systems. The program used for the terminal sessions is written in BASIC. The form of BASIC used was restricted to a set of instructions which are common to almost all dialects, and the program can be written more efficiently in more powerful dialects such as BASIC-PLUS, MITS 4.0 BASIC, or Radio Shack Level II BASIC. The program is quite small and contains fewer than 150 executable statements plus comments.

CLASSICAL CONDITIONING

Suppose a hungry dog is placed in a lab to be observed. If the dog is given food, it will salivate and begin eating. Now suppose a tone occurs before the animal is fed. At first nothing will happen. If the tone is followed by food, and this happens a number of times, the animal will begin to salivate as soon as the tone occurs. In fact, even if food does not follow the tone on a particular occurrence, the dog will still salivate when it hears the tone.

In a sense this can be thought of as meaning: "When the tone occurs food is coming." This anticipatory occurrence of salivating before receiving food indicates that the process of classical conditioning has taken place.

In the example, seeing the food is termed an unconditioned stimulus (US) and the response of salivating in the presence of the food is known as an unconditioned response (UR). Sounding the tone is termed a conditioned stimulus (CS) and the acquired (learned) response of salivating when the tone occurs is termed a conditioned response (CR).

The diagram in Figure 1 illustrates the temporal relationship between CS and US.

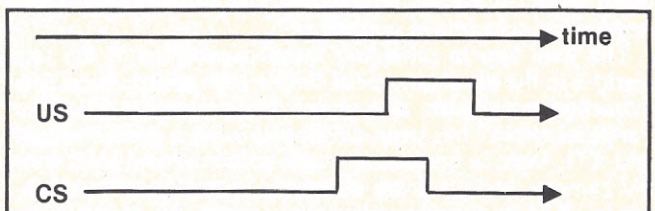


Figure 1. The temporal relationship between CS and US. The CS occurs, and then is followed by the US.

Figure 2 shows another diagram sometimes used by psychologists to illustrate both learning processes. The components of this "Learning Paradigm" are shown beneath the time line. The sequential occurrence of these components is indicated along the line. The arrows mean that a stimulus produces the response pointed to. Solid lines correspond to "Innate" association, while learned association (conditioning) is indicated by dotted lines.

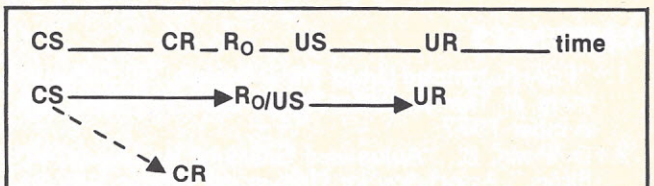


Figure 2. The components of the "learning paradigm" are shown above together with their time relationship.



PEARL:
system generation software

Let your business flow into the 1980's.

PEARL generates software tailored to specific business functions—your software problems will dissolve into solutions overnight. Accounts Payable? General Ledger? Appointment Management? PEARL generates the data fields, the edit routines, the input and output facilities.

PEARL is menu-driven and can be used by non-technicians. A novice can generate software in hours with PEARL—software that it might take a programmer months to write. Save months, or years, development time with PEARL.

PEARL is here now, for your projects. Use your time and energy to define what is needed to create business solutions and let PEARL solve the technical problems.

PEARL—Producing Error-free Automatic Rapid Logic—runs on most CP/M* based computing systems.

PEARL Level 1	$ 90	For Personal Computing
PEARL Level 2	$ 200	The Business Assistant
PEARL Level 3	$ 400	For Software Developers
PEARL Level 4	$ 800	For Turnkey System Developers

evolutionary software for the 1980's

Computer Pathways Unlimited, Inc. • 503-363-8929
2151 Davcor Street S.E. • Salem, Oregon 97302

*CP/M is a trademark of Digital Research Corp.

CIRCLE INQUIRY NO. 12

At this point, the instrumental response is shown here as R_O and is called an orienting response. The reason for this terminology will become clearer later in this section.

The slash "/" means a response is "contingent upon reinforcement." For now, this means it occurs because in the past a reward followed it. Actually, reinforcement includes both positive reinforcement (reward) and negative reinforcement (behavior which terminates a negative stimulus). This need not concern us here. However, it should be noted that negative reinforcement is not the same as punishment (the application of a negative stimulus which occurs for a certain period of time no matter what the animal does).

Another component sometimes used in the learning paradigm will be omitted because it would make this discussion unnecessarily complex. This is the goal response, S_g, which acts as a feedback on the animal's UR. (For a discussion of this topic see reference 2.)

To see how these ideas might be simulated, let's look at data from a sample terminal session in which only the classical component of the learning model is shown.

First, light (a neutral stimulus having no conditioned responses) will be presented as a conditioned stimulus (CS). Next an unconditioned stimulus (US) will be presented (e.g. food). The "innate" response (UR) it elicits will be SAL (Salivate) and is given an arbitrary value of 1.0.

Now, when the CS, LIGHT, occurs again, the effects of classical conditioning may be observed. A conditioned response of the same type as for food is elicited, but of less "STRENGTH."

STIMULUS	CONDITIONED RESPONSE	CONDITIONED STRENGTH	
LIGHT	—	—	;NO CONDITIONED ;RESPONSE PERFORMED
FOOD	SAL	1.0	
LIGHT	SAL	.01	

Next, a CR is initially conditioned to LIGHT, but on following occurrences of the CS LIGHT, no unconditioned stimulus follows the conditioned response that LIGHT elicits. This process is termed extinction (analogous to forgetting). It happens if the CS occurs, and the CR is not followed by a US. The result is that the "STRENGTH" of the CR becomes less.

STIMULUS	CONDITIONED RESPONSE	CONDITIONED STRENGTH
LIGHT	—	—
FOOD	SAL	1.0
LIGHT	SAL	.01
LIGHT	SAL	.00995
LIGHT	SAL	.00990025

When the strength goes below a certain value, it can be considered to have been extinguished, and it is as if the CR did not occur. The program could have been designed not to print a CR if its value was below a certain amount. This was not done because later a concept known as "higher order conditioning" will be examined. Demonstrating this process will involve rather small strength values, so it was decided not to ignore CR's below a given value. However, this could not only be done, but would be a way of determining when to free up a few words of memory — a potentially useful practice.

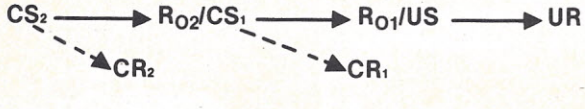


Figure 3. The illustration of higher order conditioning and secondary reinforcement.

It is also possible to learn (and forget) multiple classically conditioned responses as shown below:

STIMULUS	CONDITIONED RESPONSE	CONDITIONED STRENGTH
LIGHT	—	—
FOOD	SAL	1.0
LIGHT	SAL	.01
WATER	GULP	1.0
LIGHT	SAL	.00995
	GULP	.01

Classical conditioning is a reflexive form of learning. The animal does not have to perform a behavior which is "instrumental" in bringing about a reward. The conditioned response occurs automatically when a CS is followed by a US, and the conditioned response is much like the UR which occurs. This is not the case in instrumental conditioning, which requires problem solving type behaviors.

Classical conditioning has major consequences for several more advanced concepts relating to reinforcement and in turn upon instrumental conditioning. These ideas will be presented later, but first, let's look at how problem solving behavior is related to the process of instrumental conditioning.

INSTRUMENTAL CONDITIONING

A rat is placed in a Skinner Box (a device used to study instrumental conditioning which was designed by B.F. Skinner). In order to receive a pellet of food (reinforcement) the rat must press a bar inside the box. The response of pressing the bar to receive a reward is termed instrumental conditioning. In instrumental conditioning, the animal must perform a specific response before a reward can follow.

If a reward does follow, the instrumental response is strengthened or reinforced. The process of rewarding desired instrumental responses is termed reinforcement (The term operant conditioning may be known to the reader. This refers to an instrumental conditioning arrangement permitting the animal to respond at any time. This is sometimes called free-responding conditioning.)

Basic instrumental responses, upon which more complex ones can be built (via reinforcement), are thought to exist in the animal before conditioning. The process of conditioning simply strengthens a tendency to perform one of them. Thus moving a limb, for example, would be a basic response which could be developed into a more complex response such as pressing a bar.

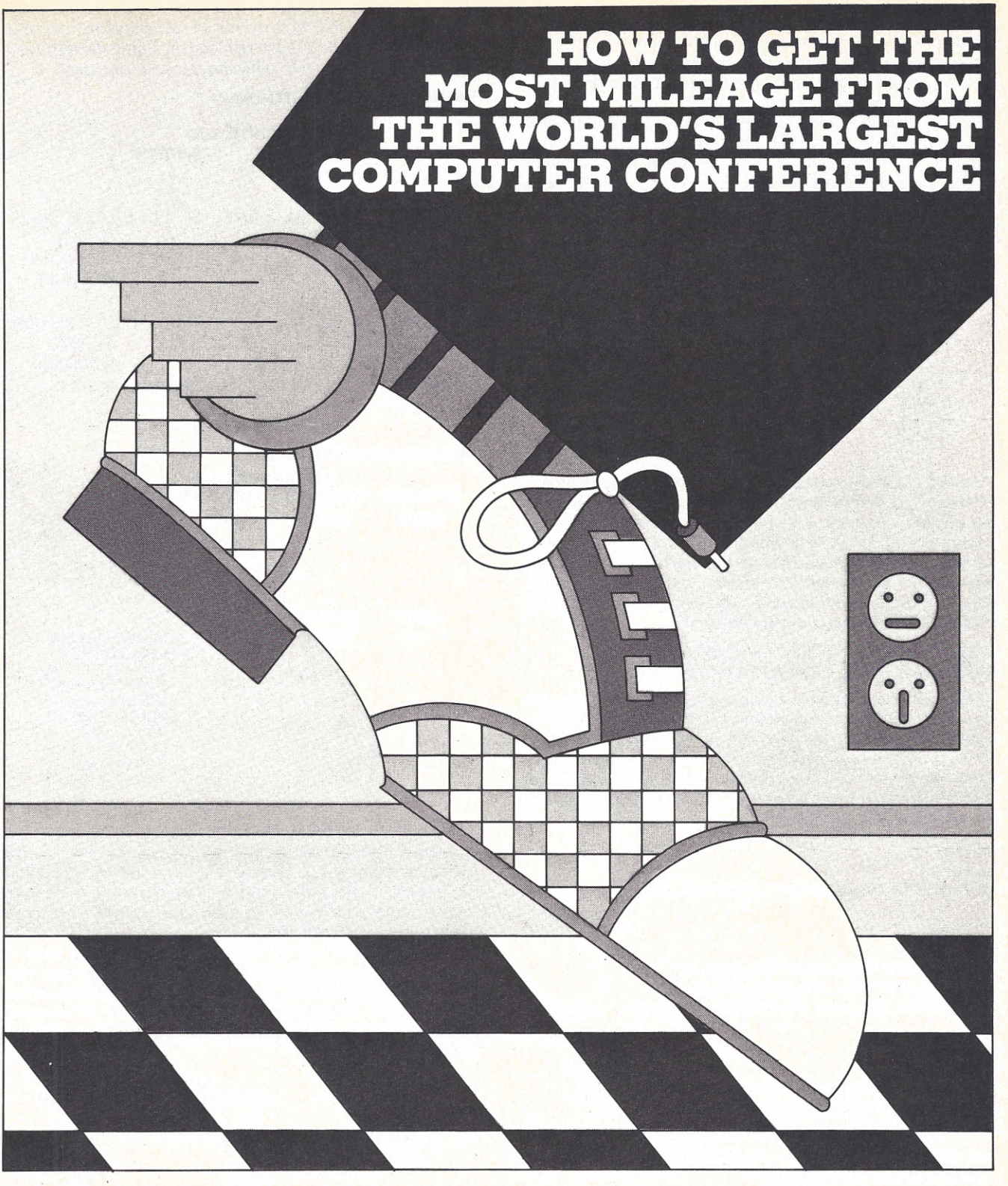
The algorithm presented here contains a number of "higher level" responses such as bar pressing. It would be possible to build these from more elementary responses, but having the complex ones facilitates the simulation, so illustration of the learning processes is more straightforward. This is a simplification which could easily be modified.

The instrumental responses are placed in a vector at the beginning of the program. Then when a particular response is desired, potential responses are performed at random. When the desired one occurs, it is reinforced. As a result of reinforcement, the desired response will now occur after the stimulus which is a cue to the computer to perform it. This is termed "shaping."

This is the process observed to occur in animals which is modeled by the learning paradigm used here. In the diagram illustrating conditioning (Figure 3), the US serves as the reinforcer and R_O is the orienting response which is "shaped" into the desired instrumental response. (The next terminal session will illustrate this.)

It is the fact that in order to receive a certain reward (or reinforcement) a certain response must be performed which

HOW TO GET THE MOST MILEAGE FROM THE WORLD'S LARGEST COMPUTER CONFERENCE



This year's National Computer Conference will feature more to see, more to learn, more to enjoy than ever. Over 1,400 exhibits...95 learning sessions...a personal computing festival...professional development seminars...a special mini-conference on computers in entertainment, and much more. Be a step ahead. Pre-register now and:

- avoid registration lines at the Conference
- save $15.00 on the full four-day program
- receive your own personal printout, highlighting areas of special interest to you
- be eligible for preferential housing

Mail in the coupon to receive your NCC '80 pre-registration forms. Step on it.

To: AFIPS, 1815 North Lynn Street, Arlington, VA 22209 I/A

☐ Please send me pre-registration, housing, and travel information forms for NCC '80.

NAME_____ CO._____

ADDRESS_____

CITY_____

STATE_____ ZIP_____

Anaheim Convention Center
May 19-22 1980 **NCC '80**

distinguishes instrumental conditioning from the automatic or "reflexive" nature of classical conditioning.

Instrumental conditioning is therefore defined as being "a form of conditioning in which the organism's response is instrumental in obtaining reinforcement."[3]

In the following session, a response occurs at random after the stimulus (the CS LIGHT). The response (PRESS) is then followed by reinforcement (the unconditioned stimulus FOOD is presented as a reward). As a result, the PRESS response strength increases and the next time LIGHT occurs, it is followed by a PRESS. After this, successive presentations of LIGHT are followed by PRESS, but there is no reinforcement. So the instrumental response begins to extinguish as happened in classical conditioning.

	INSTRUMENTAL	
STIMULUS	RESPONSE	STRENGTH
LIGHT	PRESS	.051
FOOD	—	—
LIGHT	PRESS	.05201
LIGHT	PRESS	.051485
LIGHT	PRESS	.051455

As with classical conditioning, it is possible to use reinforcement to select multiple instrumental responses. A more complex behavior consisting of a number of elementary instrumental responses could thus be learned. Since this is analogous to the terminal session illustrating multiple classically conditioned responses, no terminal session will be shown for this. Instead of conditioned responses and strengths, instrumental responses and strengths would be shown.

INTERACTION OF CLASSICAL AND INSTRUMENTAL CONDITIONING

As important as classical and instrumental conditioning are separately, the power of the learning paradigm is in the interactions between both conditioning processes.

First it should be noted that, as shown in Figure 2, both classical and instrumental conditioning are concurrent (as well as interdependent) processes. This is illustrated by the following terminal session.

```
STIMULUS
?LIGHT
INSTRUMENTAL RESPONSE AND STRENGTH PRESS    5.10000E-2

STIMULUS
?FOOD
CLASSICAL RESPONSE AND STRENGTH SAL         1
CLASSICAL RESPONSE AND STRENGTH GULP        0
INSTRUMENTAL RESPONSE AND STRENGTH ROTATES  5.10000E-2

STIMULUS
?LIGHT
CLASSICAL RESPONSE AND STRENGTH SAL         0.01
CLASSICAL RESPONSE AND STRENGTH GULP        0
INSTRUMENTAL RESPONSE AND STRENGTH PRESS    5.20100E-2
```

At this point, higher order conditioning and secondary reinforcement will be examined.

If a conditioned stimulus, CS, is initially followed with a US as shown in Figure 3, CS_1 acquires CR_1 similar to the UR. It is then possible to precede CS_1 by a neutral CS_2. The result will be that CS_2 will acquire CR_2 similar to CR_1. This is illustrated by Figure 3.

Both CR_2 and CR_1 will be of the same type of response, while the "Strength" of CR_2 will be less than the strength of CR_1.

In a sense, CS_1 is being used as a reinforcer in the same way as the US. CS_1 is termed a secondary reinforcer and it is possible to shape R_{O2} into a desired instrumental response via reinforcement with CS_1.

We now see that both forms of conditioning interrelate which can in turn be used to reinforce instrumental responses. This is illustrated by the following terminal sessions.

HIGHER ORDER CONDITIONING

	CONDITIONED		
STIMULUS	RESPONSE	STRENGTH	
LIGHT	—	—	
FOOD	SAL	1.0	
LIGHT	SAL	.01	;LIGHT HAS GAINED ;"FOODLIKE" ;QUALITIES
TONE	—	—	
LIGHT	SAL	.00995	;LIGHT IS USED ;AS FOOD WOULD BE
TONE	SAL	.0000995	;RESULT

SECONDARY REINFORCER

	INSTRUMENTAL		
STIMULUS	RESPONSE	STRENGTH	
GONG	ROTATE	.051	;ROTATE MEANS THE ;ANIMAL TURNS IN A ;CIRCLE—THIS BE- ;HAVIOR CAN ACTU- ;ALLY BE TAUGHT ;IN THIS WAY
LIGHT	NOT IMPORTANT HERE		
GONG	ROTATE	.05151	;LIGHT HAS REIN- ;FORCED THE IN- ;STRUMENTAL RE- ;SPONSE OF ROTAT- ;ING BECAUSE IT ;PRECEDED FOOD ;WHEN USED ;IN HIGHER ORDER ;CONDITIONING ;EXAMPLE ABOVE

A concrete example of this is money. These two procedures are thought to be a way of explaining how money becomes important to people. Through classical conditioning (usually higher order conditioning) money gains reinforcing properties. It may then be used as a secondary reinforcer to encourage such activities as work.

It is possible to use an unconditioned stimulus, such as FOOD, in the same place in the learning paradigm as a conditioned stimulus. (FOOD may be used as a conditioned stimulus as well as an unconditioned stimulus. All unconditioned stimuli may be thought of as being conditioned stimuli in this sense.)

	CONDITIONED	
STIMULUS	RESPONSE	STRENGTH
FOOD	SAL	1.0
WATER	GULP	1.0
FOOD	SAL	1.0
	GULP	.01

The number of instrumental responses conditioned to a given conditioned stimulus need not be static. Over a period of time, behavior may vary. New instrumental responses may be added and if reinforced these will be repeated upon future occurrences of the CS. If the new responses are not reinforced they will not occur. They did not prove to be adaptive. This effect can be noted in the following session.

```
STIMULUS
?LIGHT
CLASSICAL RESPONSE & STRENGTH:      SAL     1
CLASSICAL RESPONSE & STRENGTH:      GULP    0
INSTRUMENTAL RESPONSE & STRENGTH:   ROTATE  5.10000E-2
```

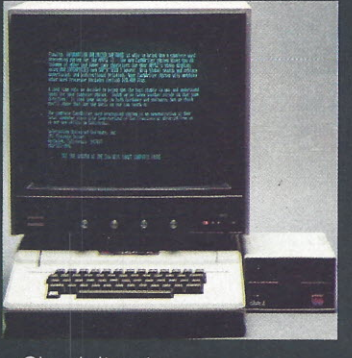


"As manager of Personal Computers, here at Computerland of San Francisco, evaluating new software products is part of my job. With all the word processors on the market today, I choose EasyWriter for my business and personal use."
—Karen Dexter Weiss

EasyWriter T.M.

80 COLUMNS OF WORD PROCESSING POWER
FOR YOUR APPLE II COMPUTER

Finally . . . INFORMATION UNLIMITED SOFTWARE is able to bring you a complete word processing system for the Apple II. The new EasyWriter system gives you *80 columns of upper and lower case characters* for your Apple's video display, using the new SUPER'R'TERM1 board!



The complete EasyWriter word processing system is on demonstration at your local computer store like Computerland of San Francisco or directly from us at our new office in California.

A long time ago, we decided to bring you the best simple-to-use and understand tools for your computer system. Today we've taken another stride in that same direction. It took some doing, in both hardware and software, but we think you'll agree that for the buck, no one can touch us.

Check it out:

- 80 Columns on the Screen!
- Upper & Lower Case!
- Global Search & Replace!
- Underlining!
- Bidirectional Printing!
- Incremental Spacing!
- File Appending!
- 50 Pages of Text Per Disk!

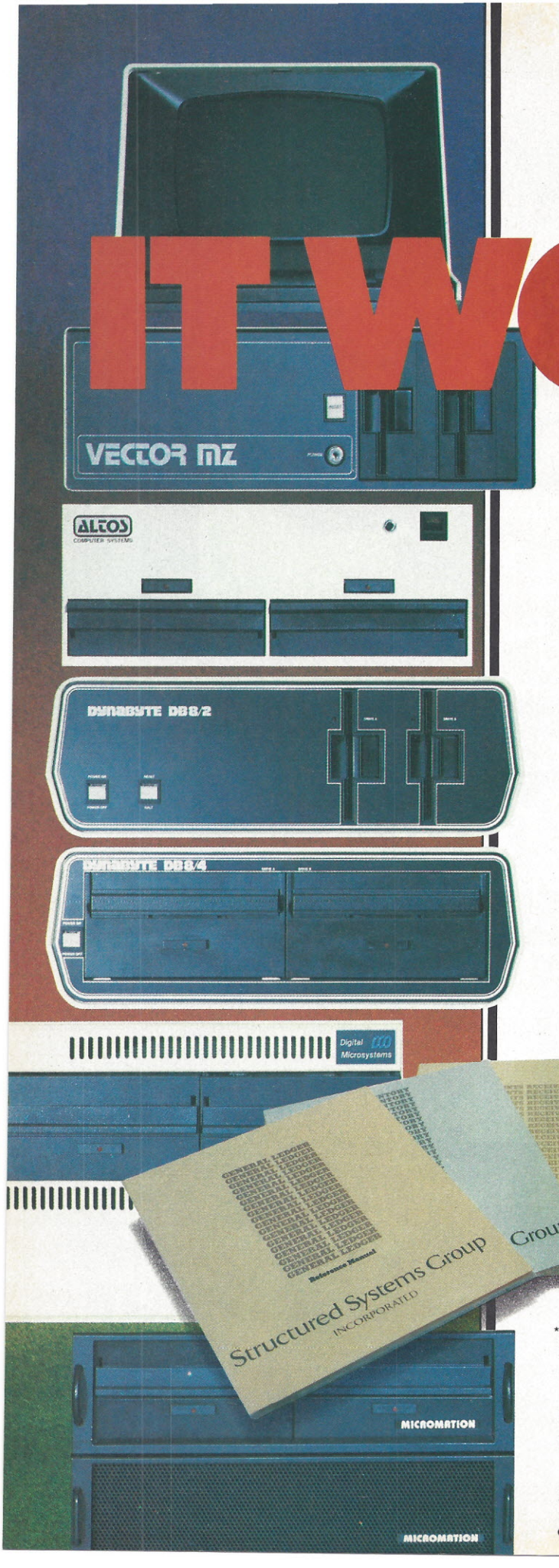
Information Unlimited Software™ Inc.

Information Unlimited Software, Inc.
793 Vincente St.
Berkeley, CA 94707
(415) 525-4046

- EasyWriter is a TM of Cap'n Software, Inc.
- Apple II is a TM of Apple Computers, Inc.

See The System at the 5th West Coast Computer Faire.

CIRCLE INQUIRY NO. 37



The presidents of the companies making these computers recommend Structured Systems Business Software for one very simple reason:

IT WORKS.

Structured Systems Business Software works for business. It works as delivered, without customization or reprogramming. It works without extensive retraining, or any new data processing personnel.

And it works hard. Our business software turns a computer into an efficient, hardworking employee, to help your business work better.

We've got working solutions for your General Ledger, Payroll, Accounts Receivable, Accounts Payable, Inventory, letter writing and correspondence, mailing lists, data storage and report writing. Ready to run on any CP/M microcomputer system.* Ready to go to work for your business right now. We help business work better, faster, and smarter.

Our Product Line Includes:

General Ledger
Accounts Payable
Inventory Control
Letteright
Accounts Receivable
Payroll
Analyst Data Storage/Report Writing
NAD Mailing System
QSORT Sort/Merge System

CP/M is a registered trademark of Digital Research.

*SSG business software runs on these and many other microcomputer systems. See your computer dealer for compatible hardware, or send for our brochure, Choosing A Computer For Your Business: SSG Software and Compatible Hardware.

Structured Systems

5204 Claremont Oakland, California 94618 (415) 547-1567

CIRCLE INQUIRY NO. 70

```
STIMULUS
?LIGHT
CLASSICAL RESPONSE & STRENGTH:     SAL      9.95000E-3
CLASSICAL RESPONSE & STRENGTH:     GULP     0
INSTRUMENTAL RESPONSE & STRENGTH:  PRESS    5.14850E-3
INSTRUMENTAL RESPONSE & STRENGTH:  ROTATE   5.1000E-2   ;New
                                                        ;instru-
                                                        ;mental
                                                        ;response
                                                        ;is tried
```

APPENDIX

OUTLINE OF THE CONDITIONING IMPLEMENTATION ALGORITHM

I. Initialization (Lines 11000 to 17400)
 Initialize variables and load possible responses;
 -load unconditional stimuli and unconditional responses associated with them
 -initialize drive strengths
 -set strength of unconditioned responses to 1.0 (this value does not change)
 -store strengths in matrix M1
 -initialize the number of conditioned stimuli to the value of M1 where the next stimulus will be put
 -initialize 1, the number of the last conditioned stimulus, to point to the first value in M1.

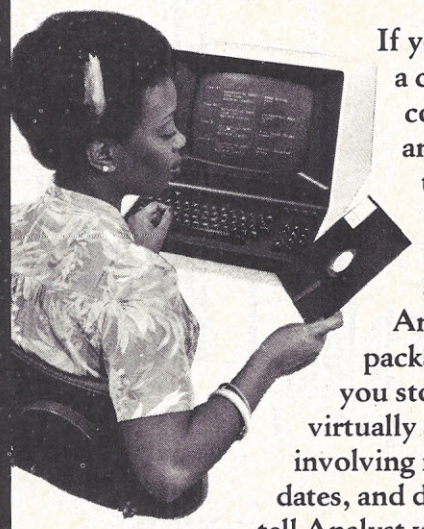
II. Read a stimulus (Lines 18000 to 18400)

III. Is the stimulus already in memory or is it a new stimulus? (Lines 18500 to 80400)
 A. Not in memory (Lines 25300 to 30500)
 -set last, L, to current, C, conditioned stimulus
 -add stimulus to memory
 -randomly try an instrumental response and pair it to stimulus
 -C = C + 1
 -Go back and read next stimulus
 B. If in memory (Lines 19400 to 25201)
 -Perform classical conditioning
 -perform classical conditioning
 -To previous stimulus add (or strengthen) CR's associated with it. The amount of strengthening is .01 * (strength of CR associated with this stimulus)
 -print CR's associated with this stimulus
 -print instrumental responses associated with this stimulus
 -try some new instrumental responses, keeping track of those tried
 -reinforcement/extinction module
 -compare strengths of CR's associated with this stimulus and the previous stimulus
 -if a strength for this stimulus is less than or equal to the one for the previous stimulus perform extinction
 -diminish strength of the CR associated with this stimulus
 -diminish strength of all instrumental responses associated with this stimulus
 -else -strengthen instrumental responses associated with previous stimulus by an increase of .01 * (their strength) + .001 * (drive strength)
 -decrease drive strength
 -get next CR
 -set L to this stimulus
 -go back and read next stimulus

REFERENCES

1. Hilgard, E.R. and Marquis, D.G., Conditioning and Learning, Appleton-Century-Crofts, 1968, pp. 436-440.
2. Hilgard, E.R. and Marquis, D.G., pp. 216-221.
3. Kendler, Howard H., Basic Psychology, Appleton-Century-Crofts, 1963, p. 691.

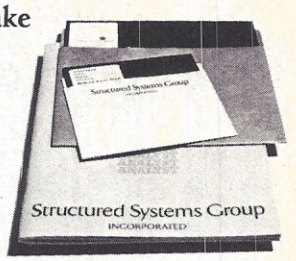
The Working Analyst.



If you would like to put a computer to work collecting, organizing, and summarizing the information you need to make better decisions, take a look at Analyst. Analyst is a software package designed to let you store and analyze virtually any information involving numbers, dollars, dates, and descriptions. Simply tell Analyst what kind of information you want to store. Analyst creates a computerized file for that information. And Analyst creates an information entry program for your file that asks you for each entry, and checks your data for errors. (You can create any number of different files.)

Then tell Analyst what reports you want from your data file. There are all sorts of record selection and report formatting options, so you can design an unlimited variety of reports to focus on different aspects of the same data file.

Analyst is so flexible, you'll find a million ways to use it. It is easy to use, so you don't need to be a programmer to make your computer really work for you. If this bit of information intrigues you, find out the rest. You'll like what you see.



Analyst is a part of a full line of working software solutions from Structured Systems Group, all ready to run on any CP/M* microcomputer system. For more information, see your computer retailer, or call us.

*CP/M is a trademark of Digital Research.

Structured Systems

5204 Claremont Oakland, Ca. 94618 (415) 547-1567

CIRCLE INQUIRY NO. 71

PROGRAM LISTING

```
1       REM
2       REM     *CONDITIONING IMPLEMENTATION ALGORITHM*
3       REM
4       REM     (AN ALGORITHM TO DESCRIBE AND SIMULATE CLASSICAL AND INSTRUMENTAL CONDITIONING)
200     REM     FOR EASE OF READING I'VE MADE ALL LABLES THAT ARE BRANCHED TO END IN 1. EXAMPLE: 18301
210     REM     TO TERMINATE A SESSION, TYPE 'FIN' WHEN A STIMULUS IS REQUESTED.
220     REM
300     REM
400     REM                             MAJOR VARIABLES AND ARRAYS:
410     REM                           C=NUMBER  OF INNATE AND CONDITIONED STIMULI
420     REM                           C1$(I)= TYPE OF CONDITIONED RESPONSE
430     REM                           C2$(I)=TYPE OF CONDITIONED STIMULUS
440     REM                           D(I)=CURRENT DRIVE STRENGTH
450     REM                           G$(I)=TYPE OF GOAL OR FEED BACK STIMULUS
460     REM                           G1(I,J)=STRENGTH OF GOAL STIMULUS
470     REM                           NOTE: THE GOAL STIMULUS WILL BE IMPLEMENTED AT A FUTURE
480     REM                                 DATE TO HELP DO PATTERN RECOGNITION.
490     REM                           M1(I,J)=STRENGTH OF CONDITIONED RESPONSE FOR A GIVEN STIMULUS
500     REM                           R$(I)=TYPE OF INSTRUMENTAL RESPONSE
510     REM                           R1(I,J)=STRENGTH OF INSTRUMENTAL RESPONSE FOR A GIVEN STIMULUS
520     REM                           U=NUMBER OF UNCONDITIONED STIMULI (INNATE STIMULI)
530     REM                           W(I)=POINTERS TO LAST INSTRUMENTAL RESPONSES
540     REM                           Y=NUMBER OF LEARNED INSTRUMENTAL RESPONSES
550     REM                           Y1=NUMBER OF POSSIBLE INSTRUMENTAL RESPONSES NOT YET LEARNED
560     REM                           Z(I)=POINTERS TO CURRENT INSTRUMENTAL RESPONSES THAT HAVE BEEN TRIED
570     REM
10000       DIM C1$(15),C2$(15),D(15),G1(15,15),G$(15),M1(15,15),R1(15,15),R$(15),W(15),Z(15)
11000   REM
11100   REM                     INITIALIZE ALL PARAMETERS TO BLANK OR ZERO
11200   REM
11300       FOR I=1 TO 15
11400         LET C1$(I)=" "
11600         LET C2$(I)=" "
11700         LET D(I)=0
11800         LET G$(I)=" "
12000         LET R$(I)=" "
12100         FOR J=1 TO 15
12200           LET M1(I,J)=0
12230           LET G1(I,J)=0
12250           LET R1(I,J)=0
12300         NEXT J
12400       NEXT I
12480   REM
12500   REM                     SET POSSIBLE INSTRUMENTAL RESPONSES WHICH MAY OCCUR
12600   REM                      N IS THE NUMBER OF POSSIBLE INSTRUMENTAL RESPONSES.
12700   REM
12800       LET N=3
12900       LET R$(1)="PRESS"
13000       LET R$(2)="JUMP"
13100       LET R$(3)="ROTATE"
15000   REM
15100   REM                     SET UNCONDITIIONED STIMULI AND THEIR ASSOCIATED RESPONSES AND THEIR
15200   REM                     ASSOCIATED DRIVE STRENGTHS. U IS THE NUMBER OF UNCONDITIONED STIMULI - THAT
15300   REM                     IS DRIVES. INNATE RESPONSES AND ASSOCIATED GOAL STIMULI ARE SET TO 1.0 AND
15310   REM                     DO NOT CHANGE. N2 IS NUMBER OF CLASSICALLY CONDITIONED RESPONSES.
15400   REM
15500       LET N2=2
15600       LET U=2
15700       LET C2$(1)="FOOD"
15800       LET C1$(1)="SAL"
15900       LET G$(1)="SALING"
16000       LET G(1)=1.0
16100       LET D(1)=.5
16200       LET M1(1,1)=1
16300       LET C2$(2)="WATER"
16400       LET C1$(2)="GULP"
16500       LET G$(2)="GULPING"
16600       LET G(2)=1.0
16800       LET M1(2,2)=1
17000   REM
17040   REM                     INITIALIZE THE NUMBER OF CONDITIONED STIMULI, C, TO THE VALUE IN THE TABLE
17080   REM                     OF THE NEXT STIMULUS.(ALL STIMULI, UNCONDITIONED AND CONDITIONED, CAN BE THOUGHT
17100   REM                     OF AS REALLY BEING CONDITIONED STIMULI.)
17140   REM                     INITIALIZE LAST CONDITIONED STIMULUS NUMBER, L, TO NUMBER OF STIMULUS OCCUR-
17180   REM                     RING PRIOR TO THE ONE WHICH WAS JUST READ IN - THAT IS TO 1.0.
17240   REM
17300       LET C=U+1
17400       LET L=1
18000   REM
18100   REM                     READ IN A STIMULUS, S$.
18200   REM
18301       PRINT
18320       PRINT "STIMULUS:"
18360       INPUT S$
18400       IF S$="FIN" GO TO 90001
18500   REM
18600   REM                     SEE IF CURRENT STIMULUS, S$, IS A CONDITIONED STIMULUS, C2$(I),
18700   REM                     ALREADY IN THE MEMORY TABLE.
18800   REM
19000       FOR K=1 TO C
19100   REM
19200   REM                     IS S$ A C2$ IN MEMORY? IF NOT GET NEXT C2$.
19300   REM
19400         IF C2$(K)<>S$ GO TO 25201
19700           FOR J=1 TO N2
19720             IF M1(L,J)=1 GO TO 21001
19730             IF L=K GO TO 21001
19740   REM
19800   REM                     NOW LAST C1$ BECOMES LIKE CURRENT CONDITIONED RESPONSE, C1$(K) THAT
19900   REM                     IS ASSOCIATED WITH CURRENT STIMULUS, S$. THIS IS DONE BY GIVING M1(L,J)
20000   REM                     A FRACTION OF M1(K,J), THE STRENGTH OF THE CONDITIONED RESPONSE THAT
20010   REM                     OCCURS WITH CONDITIONED STIMULI C2$(K).
20100   REM
20200               LET M1(L,J)=M1(L,J)+.01*M1(K,J)
20300   REM
20400   REM                     RESPOND (CLASSICAL RESPONSE) AND PICK UP FEED BACK G1(K,J).
20500   REM
21001               PRINT "CLASSICAL RESPONSE AND STRENGTH: " C1$(J),M1(K,J)
21100               LET G1(K,J)=G1(K,J)+.01*M1(K,J)
21140           NEXT J
22000   REM
22100   REM                     RESPOND (INSTRUMENTAL RESPONSE)
22200   REM
22300         LET Y=0
22320         LET Y1=0
22340   REM
22350   REM                     PRINT LEARNED INSTRUMENTAL RESPONSES. PLACE POINTERS TO THOSE
22360   REM                     LEARNED IN Z VECTOR. Y IS NUMBER OF INSTRUMENTAL RESPONSES.
22370   REM
22380         FOR J=1 TO N
22400           IF R1(K,J)>.0505 GO TO 22501
22440             LET Y1=Y1+1
22460             GO TO 22601
22501           PRINT "INSTRUMENTAL RESPONSE AND STRENGTH: " R$(J),R1(K,J)
22530           LET Y=Y+1
22560           LET Z(Y)=J
22601         NEXT J
22700   REM
22710   REM                     TRY SOME NEW RESPONSES. TRY A MAX OF HALF THE TOTAL "NOT YET LEARNED"
22720   REM                     ONES, Y1. SELECT RESPONSES AT RANDOM FROM THOSE "NOT YET LEARNED" AND
22730   REM                     GIVE THEM A SMALL INITIAL VALUE OF .051. PLACE POINTERS TO THEM IN Z
22740   REM                     VECTOR.
22750   REM
22780         IF Y1<2 GO TO 24401
22810           LET Y2=INT(Y1/2)
22820           FOR J=1 TO Y2
22830             LET X=INT(N*RND+1)
22840             IF R1(K,X)>.0505 GO TO 22891
22850               LET R1(K,X)=.051
22860               PRINT "INSTRUMENTAL RESPONSE AND STRENGTH: " R$(X),R1(K,X)
22870               LET Y=Y+1
22880               LET Z(Y)=X
22891           NEXT J
23000   REM
24000   REM                     REINFORCEMENT/EXTINCTION MODULE
24300   REM
24401         FOR J=1 TO N
24410   REM
24420   REM                     EXTINCTION (FORGETTING) OCCURS IF THE STRENGTH OF THE CURRENT,
24430   REM                     K, CONDITIONED RESPONSE IS <= THE PREVIOUS, L, ONE. IF EXTINC-
24440   REM                     TION THEN ALL INSTRUMENTAL RESPONSES ARE DECREASED IN STRENGTH,
24450   REM                     AS WELL AS STRENGTH OF CONDITIONED RESPONSE, M1(K,J), AND GOAL
24460   REM                     STIMULUS, G1(K,J).
24470   REM
24500           IF M1(K,J)>M1(L,J) GO TO 25061
24600             IF M1(K,J)=0 GO TO 25001
24700             IF M1(K,J)=1 GO TO 25001
24780               LET M1(K,J)=M1(K,J)-.005*M1(K,J)
24800               LET G1(K,J)=G1(K,J)-.005*G1(K,J)
25001             FOR J1=1 TO N
25020               IF R1(K,J1)<=.05049 GO TO 25041
25030               LET R1(K,J1)=R1(K,J1)-.0005*(M1(L,J)-M1(K,J))-.00001
25041             NEXT J1
25050           GO TO 25111
25052   REM
25053   REM                     REINFORCEMENT TAKES PLACE HERE. IF S$ IS AN INNATE (UNCONDITION-
25054   REM                     ED STIMULUS), THAT IS K<=U, OR S$ IS A SECONDARY REINFORCER
25055   REM                     THEN RESPONSES OCCURRING WITH PREVIOUS STIMULUS ARE STRENGTH-
25056   REM                     ENED. DECREASE DRIVE STRENGTH D(I). (IT'S BEEN SATISFIED.)
25057   REM
25061           FOR J1=1 TO Y3
25070             LET R1(L,W(J1))=R1(L,W(J1))+.01*R1(L,W(J1))+.001*D(K)
25080             LET D(K)=D(K)-.01*D(K)
25090           NEXT J1
25111         NEXT J
25112   REM
25113   REM                     LAST, L, BECOMES CURRENT,,K.  THOSE INSTRUMENTAL RESPONSES WHICH
25114   REM                     WERE EXECUTED ARE COPIED INTO THE VECTOR W(J) USED TO DETERMINE
25115   REM                     WHICH INSTRUMENTAL RESPONSES ARE REINFORCED.
25117   REM
25120         LET L=K
25130         FOR J=1 TO Y
25140           LET W(J)=Z(J)
25150         NEXT J
25160         LET Y3=Y
25200         GO TO 18301
25201       NEXT K
25300   REM
25400   REM                     STIMULUS NOT FOUND IN "MEMORY". IT'S A NEW ONE. IF TOO MANY STIMULI
25410   REM                     IN MEMORY, GO TO 80001 AND GRIPE ABOUT IT.
25500   REM
26000       IF C>15 GO TO 80001
30000         LET L=C
30100         LET C2$(L)=S$
30110   REM
30112   REM                     RANDOMLY TRY SOME INSTRUMENTAL RESPONSES AND COPY THE ONES TRIED
30113   REM                     INTO THE POINTER VECTOR W(J). THEN INCREASE POINTER TO NEXT STIM-
30114   REM                     ULUS, C, BY ONE.
30120   REM
30130         LET Y3=INT(N/2)
30160         FOR J=1 TO Y3
30200           LET X=INT(N*RND+1)
30230           LET R1(L,X)=.051
30260           PRINT "INSTRUMENTAL RESPONSE AND STRENGTH: " R$(X),R1(L,X)
30270           LET W(J)=X
30290         NEXT J
30400         LET C=C+1
30500         GO TO 18301
79000   REM
79100   REM                     IF EXCEED MEMORY AVAILABLE FOR CONDITIONED STIMULI, SAY SO AND IGNORE.
79200   REM
80001       PRINT "MY MEMORY IS FULL. I REFUSE TO LEARN ANY MORE."
80100       PRINT
80200       PRINT "THIS STIMULUS WAS IGNORED. LAST STIMULUS IS STILL CURRENT ONE."
80300       PRINT
80400       GO TO 18301
89700   REM
89800   REM                     ALL DONE
89900   REM
90001       PRINT "ALL FINISHED. THANKS FOR A REAL LEARNING EXPERIENCE."
99999   END
```

Can Your Robot Pass the Test?

By Alan Thompson
Robotics Age
P.O. Box 801, La Canada, CA 91011

A new international society is sponsoring prizes for robots that successfully perform everyday domestic tasks. . .

CAN YOUR ROBOT PASS THE TEST?

The domestic robot that does all the housework has long been a dream of science-fiction fans and overworked housekeepers. Yet, despite many advances in the field of industrial robotics, it seems that little is being done to work toward this goal. One hears about research robots capable of finding parts oriented at random and assembling them, but today's home robots typically can do little more than avoid bumping into walls.

One reason for this imbalance is that the talents required by a robot for successful functioning in an unstructured home environment, particularly a well-developed sense of vision, are still beyond the capabilities demonstrated by our best research robots using large mainframe computers. In addition, since the development of special-purpose techniques that work in constrained industrial environments can produce far more immediate results, considerably more effort is being applied in this area. Industry representatives have predicted that a general-purpose household robot is still decades away and will require a multi-million dollar development program.

This estimate, however, is based on the assumption that the research will be pursued in the traditional way — by professional researchers with corporate and government funding. What it fails to consider is the impact of thousands of student and amateur experimenters using increasingly powerful microcomputers to tackle these same problems. With today's micros, even the basement tinkerer has access to computer power that rivals that of many professional labs — and in many cases amateurs are more motivated and possibly more creative than some paid investigators. The problem, then, is how to focus this vast resource of talent and enthusiasm on the task of making robots smarter.

One of the difficulties faced by the would-be robot experimenter is that of obtaining knowledge about the field. Professional robotics researchers have the advantage of a thorough background in control theory and the programming techniques used in Artificial Intelligence, as well as being aware of the work done by their colleagues. This background, though it is by no means a prerequisite for doing creative work in robotics, can nonetheless save one a lot of time and lost effort.

However, the field has matured to the point that textbooks and references devoted to the subject are now available to the public, offering material that was previously available only in technical journals and conference proceedings. (See references at end of article.) Through these sources, experimenters can learn most of what they need to know to develop their own projects.

The remaining major difficulty is how to get people interested enough to spend money on building robots. Most of today's experimenters either have resources available through their schools or institutions or else get enough satisfaction out of robotics as a hobby to spend their own money on their creations. As anyone who has financed a robot project, however small, can tell you, this can easily run into thousands of dollars.

There is another way of getting people's creative energies flowing and their pocketbooks open, however. It worked in the early days of the automobile and aviation industries, and it worked most recently in the development of man-powered flight. The early land and air races and the Kremer Prize are all examples of how to get people to expand their horizons and create new technology in response to a challenge. Similarly, the prize offered by Chess Grandmaster David Levy for the first program that could beat him gave considerable impetus to the development of computer chess. So much, in fact, that Levy elected not to renew his challenge after its original term expired.

Now, finally, there is a contest for intelligent robots. The International Robotics Foundation (IRF), a recently formed non-profit organization, will be offering prizes for robots competing in a variety of categories. In conjunction with *Robotics Age Magazine*, the Foundation will hold annual trials in which robots will demonstrate their abilities in the various specialized areas of functioning needed by a household robot.

In order to allow contestants to concentrate on a particular problem, separate prizes will be awarded in each category, tentatively grouped according to manipulative skills, vision, and hearing. There will also be an award for the robot with the best performance integrating all basic skills.

Entries to the contest must be "true" robots, i.e., completely computer-controlled, with no human intervention allowed during the execution of the task. Remote-control links of any kind will not be allowed. To demonstrate the robots' ability to modify their behavior based on sensory input, no on-the-spot human programming will be allowed prior to a trial, except for program selection. The robots must be able to sense the locations of objects involved in the task.

An example task in the manipulative skills catetory would be removing dishes from a countertop and placing them in a dishwasher. Another example would be washing windows. The need for advanced sensory skills for robots competing in this category would be minimized by structuring the tasks so that they could be accomplished using only simple contact sensors or "feelers."

Robots competing in the vision category would be expected to recognize any of a number of "standard" objects. Since generalized 3-D vision (the recognition of complex solids from an arbitrary perspective) is still beyond the current state of the art in AI research, the recognition problem will be simplified by specifying in advance the perspective from which the standard objects will be viewed. This effectively reduces the problem to one of 2-D recognition, allowing the use of much simpler techniques suitable for microcomputers.

Similarly, the recognition of normal connected speech (no pauses between words) is still a difficult research problem. Isolated word recognition is much simpler, and robots competing in the hearing category may use this technique. Robot competitors must be able to follow simple verbal instructions involving movement. Scoring will be based on the number of words, and also the number of different speakers, the robot is able to recognize.

Perhaps the most exciting competition category is that of integrated robot systems. Robots in this category should have talents from all the other categories combined, although their sensory and manipulative capabilities need not be as advanced as in the robots competing in the specialized areas. The robots may be verbally asked to go find a particular kind of object (at a certain place) and bring it back, a task involving navigation around obstacles, search, recognition, and manipulation. Other sample trials would be washing floors or vacuuming around furniture.

If this sounds fantastic, remember that all of the talents needed to perform the tasks described above have been demonstrated in robotics labs around the world, and, in the estimation of the experts contacted by the Foundation, none

Radio Shack DEALER

COMPUTER SPECIALISTS



15% Discount on TRS-80's - I AND ACCESSORIES

POPULAR 16K LEVEL 2 SYSTEM.....$722.00
FAST 100 cps Centronics 730 PRINTER....$800.00
HIGHLY RELIABLE LOBO DRIVES.....$375.00

MICRO MANAGEMENT SYSTEMS
DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
912-377-7120
"TRS-80 is a Registered Trademark of Tandy Corp."

CIRCLE INQUIRY NO. 49

of the tasks are beyond the capabilities of current microprocessors, suitably applied. But that's the catch — to bring these talents out of the lab and into low-cost self-contained robots with on-board microcomputer control. That is a prerequisite to making these systems widely available for domestic use.

This annual contest will do much to bring the dream of the general-purpose household robot closer to reality. But the IRF has another plan that is guaranteed to tantalize robot experimenters around the world. Negotiations are in progress with private sponsors who will offer a substantial cash prize (probably at least $100,000) to the inventor(s) of the first universal domestic robot able to successfully perform a series of tasks outlined by the Foundation's panel of experts. The tasks would require the understanding of normal spoken sentences, generalized 3-D vision, and complex navigation and manipulative skills. Obviously, the Foundation doesn't expect to award this prize very soon, but by offering the challenge its sponsors hope to stimulate the creativity needed to make this science-fiction dream a practical reality.

Those interested in learning more about the contest and other activities of the IRF may write to:

The International Robotics Foundation
3011 Community Avenue
La Crescenta, CA 91214 □

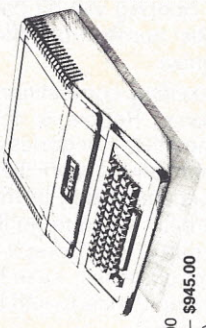
REFERENCES

"Artificial Intelligence," by Patrick H. Winston, Addison Wesley (1977). A thorough survey of AI research, explaining the theories and techniques used to make computers behave intelligently.

"How to Build a Computerized Robot," by Tod Loofbourrow, Hayden Books. A step-by-step manual on how to build an inexpensive microcomputer-controlled robot capable of sensing its environment and responding under program control.

Futra Company
P. O. Box 4380
Torrance, CA 90510
Phone (213) 370-2933

Apple II or Apple II Plus w/16K RAM



List – $1195.00
FUTRA Price – $945.00
(ship. included)

MULTI-USER! NOW! **Vector Graphics System 'B'** NEW! More RAM!



Vector MZ Microcomputer—Mindless Terminal
Boards: Z-80, 64K RAM, I/O, PROM Monitor 80 x 24 Flashwriter, 2708 PROM/RAM, Two MOD II drives, 315K each.
Software: MDOS/MBASIC, ZSM Assembler MZOS, CP/M
List Price – $5463.00
FUTRA Price – $4695.00 + shipping
***FREE with System "B":**
Complete Financial program package designed by PEACHTREE Software. G/L, A/R, A/P, Payroll + Inventory

PRINTERS: (shipping not included)

NEC "Spinwriter" 5515 R/O, RS-232
(w) Tractor, Thimble & Ribbon.
FUTRA Price $2695.00 + shipping

Texas Instruments 820 KSR, RS-232
(w) Tractor, VFC, Compressed Print
FUTRA Price $1995.00 + shipping

Integral Data "Paper Tiger" IDS-440
(w) Factory Warrantied Graphics Option.
FUTRA Price $1065.00 + shipping

Annadex DP-8000 (Standard 80 column version) or DP-8000-AP "Apple/sider" (96 Apple version).
FUTRA Price $835.00 + shipping

Trendcom 200 (80 column thermal)
High resolution graphics capability.
FUTRA Price $549.00 + shipping
Trendcom Apple-interface $75.00

APPLE ACCESSORIES: (shipping included)	**FUTRA Prices**
Disk II (w) Controller	$505.00
Disk II 2nd drive	455.00
Apple Graphics Tablet	695.00
Language System (Pascal)	397.00
Parallel Printer Int. Card	149.00
H. S. Serial Int. Card	159.00
Centronics Printer Card	179.00
Communication Card	179.00
Apple Clock Card	179.00
* ABT Apple numeric Key Pad	115.00
* ALF Apple Music Card	239.00
* D. C. Hayes Modem II	345.00
* Introl/X-10 Controller	179.00
* Corvus 10 megabyte Hard Disk Drive for Apple	4699.00
* = Independent Manufacturer	
OTHER ACCESSORIES: (shipping included)	
Leedex 12" Video Monitor	$143.00
Sanyo 9" Video Monitor	190.00
3-ring vinyl diskette sleves for 5¼" diskettes (quanity of 10)	8.00
Visi Calc (Apple Software)	120.00

FUTRA POLICY: If you find a lower price advertised on any of the items we sell, we will do our best to meet or beat that price. Terms such as shipping delivery time etc . . . must be similar. The advertised price must be in this issue or in a comparable magazine of the same month issue.

Master Charge or Visa add 2% to purchase price. No COD. Allow 3 weeks for personal checks to clear. Orders under $50.00, Add $2.00 for shipping and handling costs.
All other orders (unless specified in ad) that are within 50 lbs. limit of U.P.S. will be shipped, no charge. Purchases over 50 lbs. in weight or out of country shipped freight collect.
California residents add 6% sales tax. Phone Orders Call: (213) 370-2933

A Discussion of Artificial Intelligence

A Definitive Answer to the Question: Can Computers Think?

By Roger C. Garrett, Contributing Editor

After finishing a book on artificial intelligence that attempted to answer the question "Can computers think?" I was disappointed that the author skirted the issue by stating that we could not answer because the term thinking had not been properly defined. This is one of several books which tackle the problem of thinking computers and conclude that it cannot be answered because we don't know what thinking is. Many books in the artificial intelligence field take just that approach. Most, in fact, don't even consider it a problem. They believe it sufficient to model certain human behaviors and leave it to the reader to decide whether or not it constitutes thinking.

This has concerned me for some time and I have given considerable thought to the problem of accurately defining what we mean by "thinking." I believe that I have a workable definition. First of all it is necessary to define certain basic assumptions. I believe that one of the major stumbling blocks in the previous attempts to define thinking is the assumption that nature precludes any other entity from embodying the ability to think. I take the position that thinking can be defined without regard to the physical makeup of the thinking entity.

Another assumption often made is that human thinking is somehow linked with an extra-natural entity; that it is a special gift to humans; that it is more than the sum total of all the neurons and synapses in the brain; that there is a distinction between "mind" and "brain;" that there is some ethereal nature to thinking associated with man's special position in the scheme of things.

I tend to take a strictly scientific approach to the problem of thinking. I make no such assumptions about mankind. I want to find a description of thinking which stands by itself, which makes as few assumptions as possible. I wish to propose a definition of thinking which requires only three basic "entities." These are *Elementary Data Particles, Elementary Attributes,* and *Relationships.*

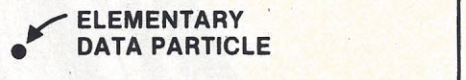


Figure 1.

Pictured here is a single *elementary data particle,* the smallest possible piece of data. Its only absolute attribute is that it exists. It has no "dimension." You could not measure it as having any particular size or shape or weight. It is comparable to the mathematical concept known as the "point."

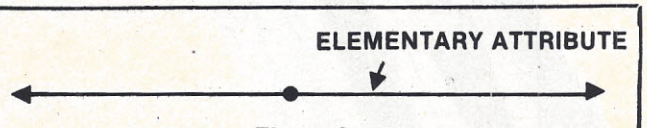


Figure 2.

We now place this elementary data particle within a "dimension," represented here as an infinitely long line. The arrows on each end of this line simply indicate that the line extends to infinity in both "directions." I put direction in quotation marks because there is no real direction indicated. This is merely a graphic representation. Indeed, the term "dimension" is simply a device for conceptually understanding the elementary attribute concept. It is not to be confused with any of the four common dimensions of length, height, width, and time. It is used here as a graphic representation of elementary attribute.

We can think of the dimension and elementary data particle as constituting a "system." Furthermore we assume that within this system all that exists is the dimension and the particle. There is no "unit of measure" or "origin" in this system so, although we can correctly state that the particle exists within the dimension, we cannot say that it exists at some known position within the dimension.

What we are saying here is that the attribute which we would call "position" within the dimension is actually "relative position." In a system with only one particle existing within a dimension (and this would hold true no matter how many dimensions we have) there is nothing for that particle to relate to so there is no position attribute.

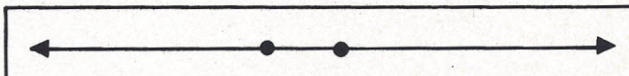


Figure 3.

Now suppose that we add a second elementary data particle. Does there now exist a relative position attribute, i.e. the *elementary attribute*? If we were sitting on one of the particles, could we describe where we are? No. There is still no origin so we cannot describe our position relative to an origin.

Can we describe our position in terms of the position of the other particle? With no unit length we are not able to say how far we are from the other particle. We also cannot say that we are left-of or right-of the other particle since there is no left or right in this system. You and I, looking at this diagram, can surely perceive left and right; but from *within* the system consisting only of the dimension and the two particles there is no left or right. So there is still no position attribute.

Also realize that since a data particle is uniquely defined by its attributes, if there exist two data particles with identical attributes then the two are not uniquely defined, they are identical. There actually exists only *one* such particle. It therefore is not possible to have coexistent (i.e. occupying the same relative position within a dimension) particles.

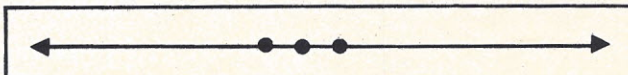


Figure 4.

We have now added a third particle. Do the particles now have a relative position attribute? Since the spacing between adjacent particles are equal, is there any way of specifying that one particle is a certain distance from another? Again the answer is no, because there is no unit length standard. No matter how many elementary particles we add to this world, provided they are spaced evenly, we do not have a relative position attribute. This is a very special situation and we will call it a *singularity* because there is no way of differentiating between the several data particles.

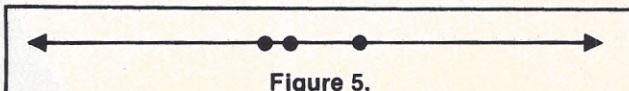
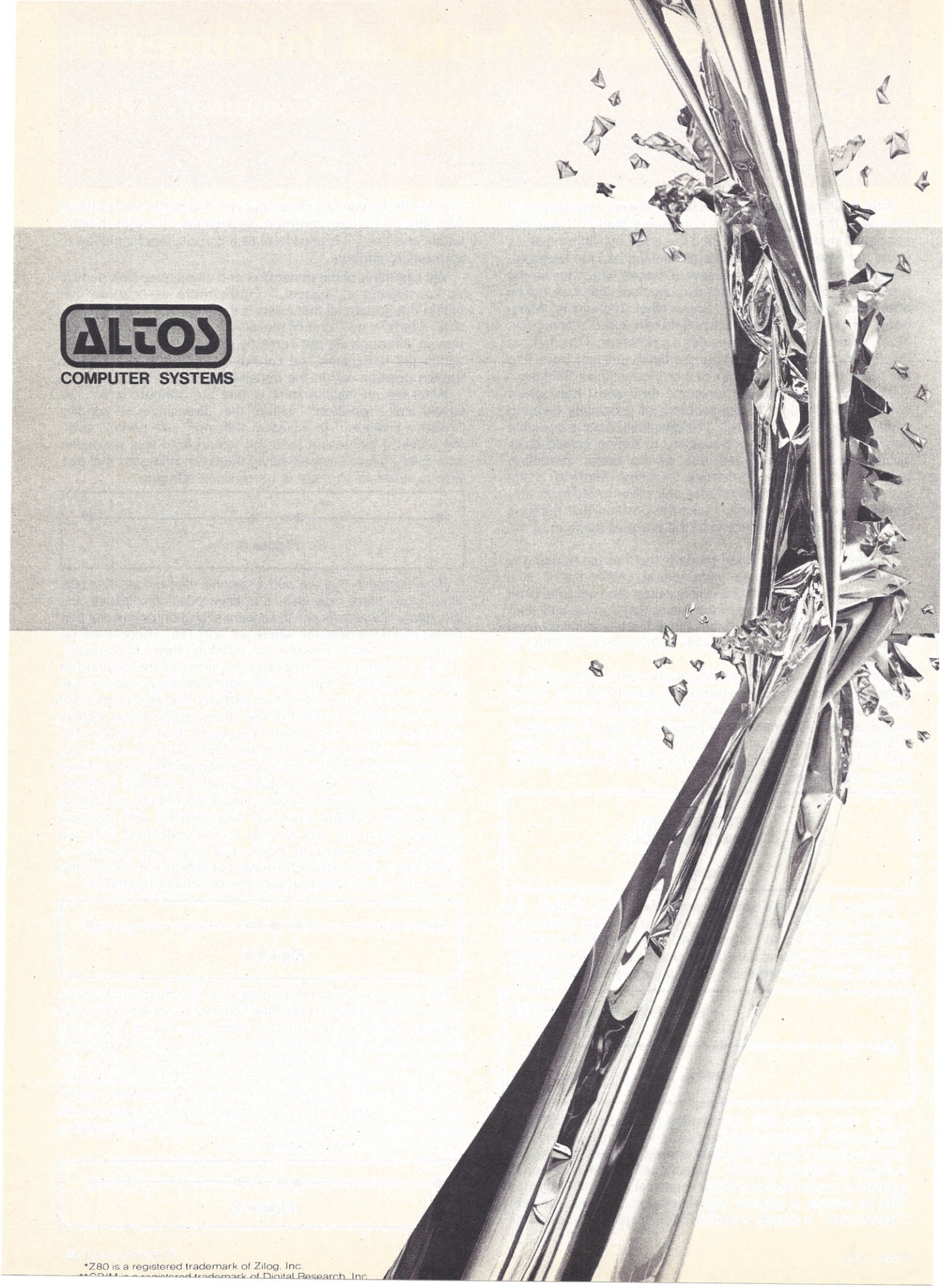


Figure 5.



ALTOS
COMPUTER SYSTEMS

*Z80 is a registered trademark of Zilog, Inc.
**CP/M is a registered trademark of Digital Research, Inc.

ALTOS BREAKS THE MICRO BARRIER.

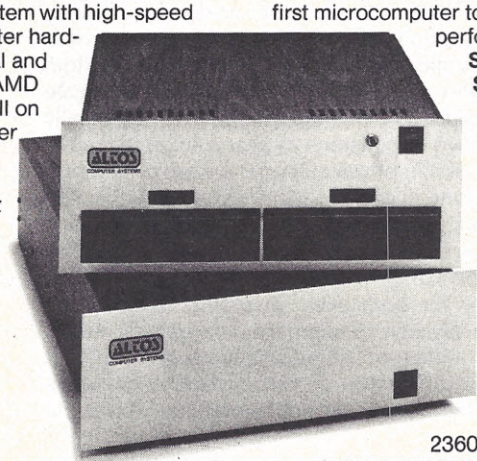
Yesterday, microcomputer meant micro performance. Once you outgrew it, you had to step up to a mini. Which meant a big step up in price.

Today, there's the new Altos ACS8000-6 single-board microcomputer system.

It's the first system for the OEM, small businessman and personal user, that offers minicomputer performance and minicomputer storage capacities—at a microcomputer price.

MULTI-USER, WINCHESTER STORAGE, FLOPPY BACK UP: $14,260.

The new Altos ACS8000-6 is a highly advanced Z80* based microcomputer system with high-speed RAM, floppy disk and Winchester hard-disk controllers, DMA, six serial and two parallel I/O ports and the AMD 9511 floating point processor all on a single board. A typical four-user system configuration with two megabytes of Shugart floppy and 29.0 megabytes of Shugart Winchester storage, including CPU and 208K bytes of RAM, costs only $14,260—compared to $30,000 or more for a similar minicomputer system. And that adds up to mini performance at less than half the cost!



MULTI-USER EXECUTIVE SUPPORTS FOUR INDEPENDENT USERS RUNNING CP/M COMPATIBLE PROGRAMS.**

This revolutionary new microcomputer system features an Altos-developed Multi-User Executive (AMEX) software program that's unique in two ways. It includes a multi-user CP/M capability and the ability to handle Winchester-type hard disks. This advanced Z80 operating program supports four independent CP/M compatible programs in any of six popular languages: BASIC, FORTRAN, COBOL, PASCAL, APL, C, and a large assortment of additional business application packages. AMEX is compatible with both the 1.4 and 2.0 versions of Digital Research's CP/M, which means programs based on either version can run under AMEX without modification.

With AMEX at the helm, your Altos ACS8000-6 system can support up to four simultaneous users with 48K bytes of RAM each plus 58 megabytes of Winchester storage and 4 megabytes of floppy back up. And that adds up to the first microcomputer to give you the power and performance of a minicomputer.

SINGLE-USER, HARD-DISK SYSTEMS START AT $9450.

The Altos ACS8000-6 series. It's a barrier breaker in every sense. Our entry-level, single-user, hard-disc system with floppy back up is priced under $10,000 and even our 4-user CP/M model is available for under $12,000. And all configurations are easily upgraded. For specific details about pricing or performance, call or write: Altos Computer Systems, 2360 Bering Drive, San Jose, CA (408) 946-6700. TELEX 171562 ALTOS SNA.

ALTOS
COMPUTER SYSTEMS

CIRCLE INQUIRY NO. 2

Now assume that the distances are not equal. We can finally describe their positions, although never in absolute terms. The position of each elementary data particle can be described in terms of percentages of the distances between other particles. The dimension represents an attribute of the data particles and the relative "positions" of the data particle represent the individual "values" of the attribute for each particle.

This constitutes an *information state.* Why? Because I define information state as being a set comprised of a set of attribute dimensions, a set of elementary data particles, and a set of relationships which define the "relative positions" of the elementary data particles within the set of attribute dimensions. In the above "system" we have one attribute dimension, three elementary data particles, and a set of relationships which define their positions, in this case expressed in terms of relative distances between particles.

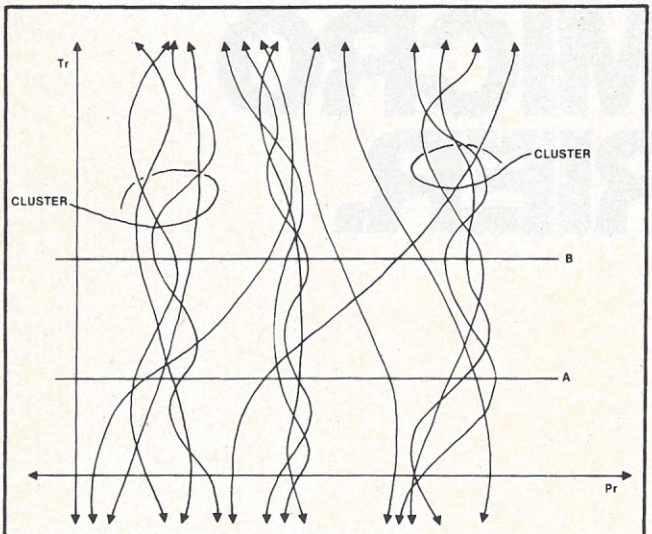


Figure 6.

In Figure 6 I have added quite a bit more to our system. I have labelled the one dimension with Pr which stands for relative Position. More importantly, I have added a second dimension, labelled Tr. For the moment I am not going to say what that stands for, but you should soon be able to figure it out for yourself.

As with the Pr dimension, the Tr dimension has no absolute unit of length. The arrows on either end simply signify that the dimension is infinite. The junction of the two dimension lines has no significance. It does *not* indicate an origin of a coordinate system. The dimension lines do *not* indicate axis. They merely represent two dimensions. Conceptually we may think of the two dimensions as specifying a plane but we should not think of it in physical terms; we are only representing two dimensions in the broadest sense of the word.

We have added a few more elementary data particles and have increased the set of relations which define the relative positions. More importantly, these relationships now define the relative positions for *both* axes. In mathematical terms the relative position of each elementary data particle within the Pr dimension is a function of the relative position of that particle within the Tr dimension. In our pictorial representation of this world, the elementary data particles appear to trace out lines.

If we draw a horizontal line through the data particle traces, we delimit an individual information state. Remember that an information state is defined as a set of elementary data particles and their relative positions along the Pr attribute axis. Lines A and B in Figure 6 indicate two discrete information states. Now here comes the important part: *thinking* is defined as *the transition from one information state to another information state.*

This "system" can be made more complex by either increasing the number of elementary data particles or by increasing the relationships, defined in terms of Pr and Tr. Notice that, as the particle traces move from the lower portion of the world to the upper portion, some of the particles begin to *"cluster."* They begin to interact as a unit, as though there is a higher level relationship which defines the relative position of the cluster. The cluster essentially becomes a data particle in its own right and several lowest level relationships have combined to form a higher level relationship. As more and more clustering occurs, the complexity of the system increases. High level relationships grow out of low level relationships. Higher level data grows from lower level data.

At the lowest level each elementary data particle "experiences" the other data particles on a very low level. We define "experiencing" as meaning "being affected by." The relative position of each particle is affected by the position of some set of the remaining particles. As clustering occurs, the higher level data (the clusters) experience not only the lower level influences but also the higher level influences, i.e. the higher level relationships. Elementary relationships breed cluster relationships. Cluster relationships breed higher level cluster relationships. And so on.

This higher level of relationships can be thought of as adding attributes to the system. The higher level of interaction (relationships) effectively defines a higher level (or at least additional) *attribute/dimension.* So, although there actually are only two dimensions, Pr and Tr, the higher level relationships cause the world to act as though there were more.

The elementary data particles only have one attribute, that of relative position at a given information state. As clustering occurs, however, the clusters acquire higher level attributes. What this is saying is that attributes are related to (functions of) the set of elementary particles and the set of elementary relationships.

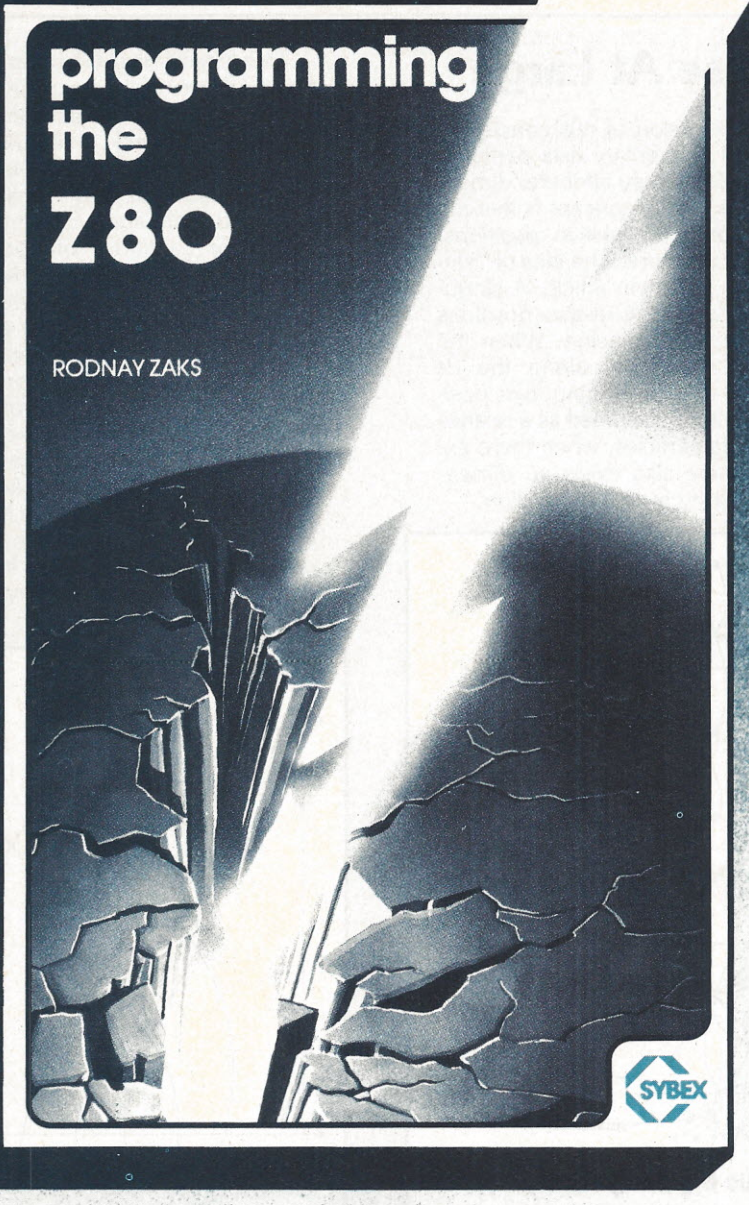
Let's look at this in terms we are more familiar with. Assume we have some system of particles, relationships, and the two elementary attribute/dimensions (Tr and Pr). The particles begin to cluster as they move along the Tr dimension. Clusters begin to cluster, relationships increase and become more complex.

At some level of complexity certain clusters begin to experience other clusters. This experience takes the form of experiencing high level attributes such as size, mass, charge, and spin. In other words, as electrons. Other clusters acquire attributes which we would recognize as protons, others are experienced as neutrons, and so on. These clusters combine to be experienced as the multitude of atoms in what you and I consider the "real world." Atoms cluster to form molecules. Molecules cluster to form planets, stars, quasars, and black holes. And some clusters even reach the level of complexity which we experience as "living creatures." Every "real world" entity is actually a specific level of clustering, a particular level of complexity.

Now if we wish to discuss whether or not computers can think, at least we have a clear concept of what we mean by thinking. If we can agree upon this definition (and admittedly that is an assumption of tremendous importance) we can then logically and rationally discuss the question of thinking machines without appealing to an ethereal assumption that man is somehow special and therefore *must* be the only entity capable of thinking.

Yet we still have not fully addressed the question. I have defined thinking as the transition from one information state to another where information state is taken to mean a set of data and the relative attributes of those data. This definition clearly defines the process which goes on inside the human brain. In the brain we have a highly complex set of chemicals, organized according to well-defined (albeit ill-understood) patterns into structures which we know as

POWER!



now! a step by step approach making your Z80 work for you

A comprehensive introduction to programming the Z80 ranging from basic concepts to addressing techniques, internal operation and advanced data structures. Exercises check your progress at each step.
624 pp., Ref. C280 $14.95.

CIRCLE INQUIRY NO. 74

TO ORDER:
By Phone: (415) 848-8233, Visa, MC, Am Ex
By Mail: Indicate quantity desired. Pre-payment required
Shipping: Add $1.50 per book (UPS) or 75¢ (4th Class–allow 4 weeks delivery)
Tax: In California please add tax

SYBEX

Dept. IA 4
2344 Sixth St.
Berkeley, CA 94710

Please send me_____ Charge my____Visa:____Mastercharge
Name_____ ____American Express
Company_____ Card #_____Exp. Date____
Address_____ Signature_____
City_____State____Zip____ ☐ Please send me your detailed catalog.

AVAILABLE AT BOOK AND COMPUTER STORES EVERYWHERE

neurons. We can choose among any number of levels of complexity to define the brain but for our discussion we can take the neuron as the basic data particle.

Each such neuron has some set of attributes, including size, electrical and chemical state, synapse composition, and so on. The relationships between those neurons are defined by the multitude of connections between neurons. As we progress through time the attributes of the neurons change, particularly the chemical and electrical states, defined by these inter-neuron relationships. So we see that the brain does indeed think (by our definition).

But that wasn't the question. We want to know whether a *computer* can think.

So let's look at what a computer can do. More precisely, let's look at what a computer *program* can do. Every program has two essential parts: data and instructions. The data have attributes, often expressed as its current numerical (or coded) value. The instructions, or set of executable statements, defines the relationships between the data. Since most programs consist of a set of *sequentially executed* instructions there is actually an additional datum which we may call the *clock.* The running of the program may then be represented as a graph such as the one shown in Figure 7.

In the upper portion of the figure we have a simple program written in BASIC. The "clock" datum mentioned above is represented by the line numbers 5, 10, 15, etc. This set of statements defines an initial information state (all data variables, including the "clock") attributes (values) initially set to zero and the relationships among these data. The program defines the transition from one information state (initial values) to another information state (the final values). When the program is executed, the transition between infor-

The Universe At Large

There is an interesting extension of this concept of defining systems as sets of elementary data particles, relationships, and the two elementary attribute/dimensions. An important feature of this concept is that the dimensions are defined only as relative positions. There is no origin or unit of measure. The idea of "singularity" was introduced in the main article. A singularity was defined in reference to the relative positions of the data particles in the Pr dimension. When the data particles are evenly distributed along the Pr dimension then there is no way of defining their positions. Since the dimension itself is defined as a relative position attribute of the data particles, when there exists no relative positions there also exists *no dimension.* It reduces to a singularity in that dimension.

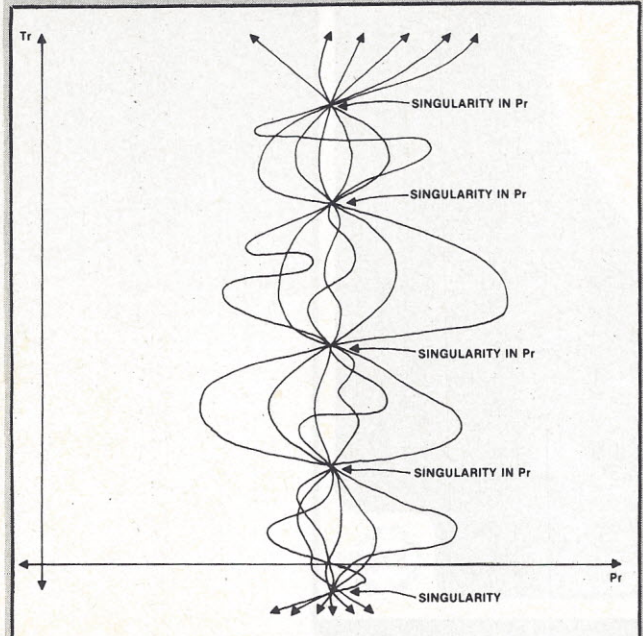


Figure 1a. Cyclic Big Bang Theory

Now consider Figure 1a where we have both the Pr and Tr dimensions. We assume a set of particles with some defined initial information state at A, and a set of relationships which define the relative positions of the particles in both the Pr and Tr dimensions. We know that we would get a singularity if at some point along the Tr dimension all of the particles become evenly distributed.

This type of singularity can be compared to the "Big Bang" theory of the universe. This theory states that all matter was, at some distant time in the past, compressed into one tiny speck, a singularity, and that it exploded to form the universe. An extension of this theory states that eventually the universe will stop expanding and fall back into itself, again forming the singularity, and that this cycle continues on forever. In our Pr and Tr system this means that periodically the data particles form a singularity in Pr. This means that periodically the Pr dimension ceases to exist.

But remember, *both* dimensions are relative dimensions. This means that Tr only exists as a function of the *transition* from one information state to another, hence the label Tr for relative Transition. If at any point along the Tr dimension the information state at that state is identical to the very next information state then there is no transition and we have a singularity in Tr. In other words, at that point in the Tr dimension *the Tr dimension does not exist.* Indeed, the Tr dimension cannot exist anywhere beyond that point. There simply is no "beyond that point."

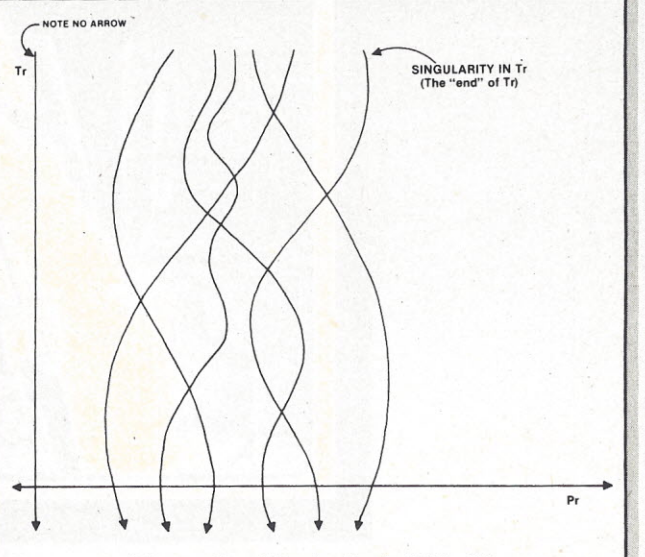


Figure 2a. Single Ended World

Figure 2a shows a world with one singularity in Tr. Pictorially this means it has one "end" and extends infinitely far in the other direction. Figure 3a depicts a world system with two singularities in Tr, meaning that the system is finite in Tr (although it is infinite in Pr).

Continued on Page 92

CP/M* VERSION 2 FOR TRS-80★ MODEL II NOW AVAILABLE

Prices reflect distribution on 8" single density diskettes. If a format is requested which requires additional diskettes, a surcharge of $8. per additional diskette will be added.

All Lifeboat programs require CP/M, unless otherwise stated.

DIGITAL RESEARCH

Software with Manual / Alone

☐ **CP/M FLOPPY DISKETTE OPERATING SYSTEM** — Ⓜ Packages supplied on diskette complete with 8080 assembler, text editor, 8080 debugger and various utilities plus full documentation. CP/M available configured for most popular computer/disk systems including: North Star Single, Double or Quad density, Altair 8" disks, Helios II, Exidy Sorcerer, Vector MZ, Heath H17† or H89†, TRS-80†, iCOM 3712 and iCOM Micro Disk plus many other configurations available off the shelf **$145/$25**
CP/M version 2 (not all formats available immediately) **$170/$25**

☐ **MP/M*** **$300/$50**

Lower prices!

☐ **MAC** — 8080 Macro Assembler. Full Intel macro definitions. Pseudo Ops include RPC, IRP, REPT, TITLE, PAGE, and MACLIB. Z80 library included. Produces Intel absolute hex output plus symbols file for use by SID (see below) **$85/$15**

☐ **SID** — 8080 symbolic debugger. Full trace, pass count and break-point program testing system with backtrace and histogram utilities. When used with MAC, provides full symbolic display of memory labels and equated values **$70/$15**

NEW!

☐ **ZSID** — As above for Z80. Requires Z80 CPU **$95/$25**

☐ **TEX** — Text formatter to create paginated, page-numbered and justified copy from source text files, directable to disk or printer **$70/$15**

☐ **DESPOOL** — Program to permit simultaneous printing of data from disk while user executes another program from the console **$45/$5**

All Microsoft prices are discounted!

MICROSOFT

☐ **BASIC-80** — Disk Extended BASIC, ANSI compatible Ⓛ with long variable names, WHILE/WEND, chaining, Ⓜ variable length file records **$300/$25**

☐ **BASIC COMPILER** — Language compatible with Ⓛ BASIC-80 and 3-10 times faster execution. Produces Ⓜ standard Microsoft relocatable binary output. Includes Macro-80. Also linkable to FORTRAN-80 or COBOL-80 code modules **$350/$25**

☐ **FORTRAN-80** — ANSI 66 (except for COMPLEX) plus Ⓛ many extensions. Includes relocatable object com- Ⓜ piler, linking loader, library with manager. Also includes MACRO-80 (see below) **$400/$25**

☐ **COBOL-80** — Level 1 ANSI '74 standard COBOL plus Ⓛ most of Level 2. Full sequential, relative, and in- Ⓜ dexed file support with variable file names. STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 **$625/$25**

☐ **MACRO-80** — 8080/Z80 Macro Asembler. Intel and Ⓛ Zilog mnemonics supported. Relocatable linkable Ⓜ output. Loader, Library Manager and Cross Reference List utilities included **$149/$15**

☐ **XMACRO-86** — 8086 cross assembler. All Macro and Ⓛ utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available **$275/$25**

☐ **EDIT-80** — Very fast random access text editor for text Ⓛ with or without line numbers. Global and intra-line commands supported. File compare utility included. **$89/$15**

MICRO FOCUS

☐ **STANDARD CIS COBOL** — ANSI '74 COBOL stand- Ⓛ ard compiler fully validated by U.S. Navy tests to ANSI level 1. Supports many features to level 2 including dynamic loading of COBOL modules and a full ISAM file facility. Also, program segmentation, interactive debug and powerful interactive extensions to support protected and unprotected CRT screen formatting from COBOL programs used with any dumb terminal **$850/$50**

☐ **FORMS 2** — CRT screen editor. Output is COBOL data Ⓛ descriptions for copying into CIS COBOL programs. Automatically creates a query and update program of indexed files using CRT protected and unprotected screen formats. No programming experience needed. Output program directly compiled by CIS COBOL (standard) **$200/$20**

☐ **HDBS** — Hierarchical Data Base System. CODASYL oriented with FILEs, SETs, RECORDs and ITEMs which are all user defined. ADD, DELETE, UPDATE, SEARCH, and TRAVERSE commands supported. SET ordering is sorted, FIFO, LIFO, next or prior. One to many set relationship supported. Read/Write protection at the FILE level. Supports FILEs which extend over multiple floppy or hard disk devices.

☐ **MDBS** — Micro Data Base System. Full network data base with all features of HDBS plus multi-level Read/Write protection for FILE, SET, RECORD and ITEM. Explicit representation of one to one, one to many, many to many, and many to one SET relationships. Supports multiple owner and multiple record types within SETs. HDBS files are fully compatible.

☐ **MDBS-DRS** — MDBS with Dynamic Restructuring System option which allows altering MDBS data bases when new ITEMs, RECORDs, or SETs are needed without changing existing data.

HDBS-Z80 version **$250/$35**
MDBS-Z80 version **$750/$35**
MDBS-DRS-Z80 version **$850/$35**
8080 Version available at $75. extra.

Z80 version requires 20K RAM. 8080 version requires 24K RAM. (Memory requirements are additional to CP/M and application program.)

When ordering HDBS or MDBS please specify if the version required is for 1) Microsoft L80 i.e. FORTRAN-80, COBOL-80, BASIC COMPILER, 2) MBASIC 4. XX, or 3) BASIC-80 5.0.

Prices and specifications subject to change without notice.

Lower prices!

EIDOS SYSTEMS

Software with Manual / Alone

☐ **KISS** — Keyed Index Sequential Search. Offers com- Ⓛ plete Multi-Keyed Index Sequential and Direct Access file management. Includes built-in utility functions for 16 or 32 bit arithmetic, string/integer conversion and string compare. Delivered as a relocatable linkable module in Microsoft format for use with FORTRAN-80 or COBOL-80, etc. **$335/$23**

☐ **KBASIC** — Microsoft Disk Extended BASIC with all Ⓛ KISS facilities, integrated by implementation of nine additional commands in language. Package includes KISS. REL as described above, and a sample mail list program **$585/$45**
To licensed users of Microsoft BASIC-80 (MBASIC) **$435/$45**

All Micropro prices are discounted!

MICROPRO

☐ **SUPER-SORT I** — Sort, merge, extract utility as abso- Ⓛ lute executable program or linkable module in Micro- Ⓜ soft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating, fixed point, exponential, field justified, etc. Even variable number of fields per record! **.$225/$25**

☐ **SUPER-SORT II** — Above available as absolute pro- Ⓛ Ⓜ gram only **$175/$25**

☐ **SUPER-SORT III** — As II without SELECT/EXCLUDE Ⓛ Ⓜ **$125/$25**

☐ **WORD-STAR** — Menu driven visual word processing Ⓛ system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, Ⓜ page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning **$445/$40**

☐ **WORD-STAR/MAIL-MERGE** — As above with option Ⓛ for production mailing of personalized documents Ⓜ with mail list from Datastar or NAD **$575/$25**

☐ **WORD-STAR Customization Notes** — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR **NA/$100**

☐ **WORD-MASTER** Text Editor — In one mode has super- Ⓛ set of CP/M's ED commands including global search- Ⓜ ing and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal **$125/$25**

☐ **DATASTAR** — Professional forms control entry and Ⓛ display system for key-to-disk data capture. Menu Ⓜ driven with built-in learning aids. Input field verification by length, mask, attribute (i.e. uppercase, lowercase, numeric, auto dup., etc.). Built-in arithmetic capabilities using keyed data, constants and derived values. Visual feedback for ease of forms design. Files compatible with all CP/M-MP/M supported languages. Requires 32K CP/M **$350/$35**

☐ **CBASIC-2** Disk Extended BASIC — Non-interactive Ⓜ BASIC with pseudo-code compiler and runtime interpreter. Supports full file control, chaining, integer and extended precision variables, etc. **$109/$15**

☐ **PASCAL/M** — Compiler generates P code from ex- Ⓘ tended language, implementation of standard PAS- Ⓛ CAL. Supports overlay structure through additional procedure calls and the SEGMENT procedure type. Provides convenient string handling capability with the added variable type STRING. Unpyped files allow memory image I/O. Requires 56K CP/M **$350/$30**

☐ **PASCAL/Z** — Z80 native code PASCAL compiler. Pro- Ⓘ duces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, companion macro-assembler and source for the library. Requires 56K and Z80 CPU.
Version 2 includes all of Jensen/Wirth except variant records **$275/$25**
Version 3 Upgrade with variant records and strings expected 3/80 **$395/$25**

☐ **PASCAL/MT** — Subset of standard PASCAL. Gener- Ⓜ ates ROMable 8080 machine code. Symbolic debug- Ⓧ ger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Version 3 includes Sets, Enumeration and Record data types. Manual explains BASIC to PASCAL conversion. Source for the run time package requires MAC (See under Digital Research). Requires 32K **$250/$30**

New Version New Features

STRUCTURED SYSTEMS GROUP

☐ **GENERAL LEDGER** — Interactive and flexible system †† providing proof and report outputs. Customization of COA created interactively. Multiple branch accounting centers. Extensive checking performed at data entry for proof, COA correctness, etc. Journal entries may be batched prior to posting. Closing procedure automatically backs up input files. Now includes Statement of Changes in Financial Position. Requires CBASIC-2 **$1250/$25**

☐ **ACCOUNTS RECEIVABLE** — Open item system with †† output for internal aged reports and customer-oriented statement and billing purposes. On-Line Enquiry permits information for Customer Service and Credit departments. Interface to General Ledger provided if both systems used. Requires CBASIC-2. **$1250/$25**

☐ **ACCOUNTS PAYABLE** — Provides aged statements †† of accounts by vendor with check writing for selected invoices. Can be used alone or with General Ledger and/or with NAD. Requires CBASIC-2 **$1250/$25**

☐ **PAYROLL** — Flexible payroll system handles weekly, †† bi-weekly, semi-monthly and monthly payroll periods. Tips, bonuses, re-imbursements, advances, sick pay, vacation pay, and compensation time are all part of the payroll records. Prints government required periodic reports and will post to multiple SSG general ledger accounts. Requires CBASIC-2 **$1250/$25**

☐ **INVENTORY CONTROL SYSTEM** — Performs control †† functions of adding and depleting stock items, adding new items and deleting old items. Tracks quantity of items on hand, on order and back-ordered. Optional hard copy audit trail is available. Reports include Master Item List, Stock Activity, Stock Valuation and Re-order List. Requires CBASIC-2 **$1250/$25**

☐ **ANALYST** — Customized data entry and reporting sys- †† tem. User specifies up to 75 data items per record. Interactive data entry, retrieval, and update facility makes information management easy. Sophisticated report generator provides customized reports using selected records with multiple level break-points for summarization. Requires CBASIC-2 **$250/$15**

☐ **LETTERIGHT** — Program to create, edit and type let- †† ters or other documents. Has facilities to enter, display, delete and move text, with good video screen presentation. Designed to integrate with NAD for form letter mailings. Requires CBASIC-2 .. **$200/$25**

☐ **NAD** Name and Address selection system — interac- †† tive mail list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. Requires CBASIC-2 **$100/$20**

☐ **QSORT** — Fast sort/merge program for files with fixed †† record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created **$100/$20**

GRAHAM-DORIAN SOFTWARE SYSTEMS

☐ **GENERAL LEDGER** — An on-line system; no batch- Ⓛ ing is required. Entries to other GRAHAM-DORIAN Ⓜ accounting packages are automatically posted. User †† establishes customized C.O.A. Provides transaction register, record of journal entries, trial balances and monthly closings. Keeps 14 month history and provides comparison of current year with previous year. Requires CBASIC-2. Supplied in source ... **$995/$35**

New Features

☐ **ACCOUNTS PAYABLE** — Maintains vendor list and Ⓛ check register. Performs cash flow analysis. Flexible Ⓜ — writes checks to specific vendor for certain in- †† voices or can make partial payments. Automatically posts to GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source **$995/$35**

☐ **ACCOUNTS RECEIVABLE** — Creates trial balance re- Ⓛ ports, prepares statements, ages accounts and rec- Ⓜ ords invoices. Provides complete information describ- †† ing customer payment activity. Receipts can be posted to different ledger accounts. Entries automatically update GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source **$995/$35**

☐ **PAYROLL SYSTEM** — Maintains employee master file. Ⓛ Computes payroll withholding for FICA, Federal and Ⓜ State taxes. Prints payroll register, checks, quarterly †† reports and W-2 forms. Can generate ad-hoc reports and employee form letters with mail labels. Requires CBASIC-2. Supplied in source **$590/$35**

New Price!

☐ **INVENTORY SYSTEM** — Captures stock levels, costs, Ⓛ sources, sales, ages, turnover, markup, etc. Trans- Ⓜ action information may be entered for reporting by †† salesman, type of sale, date of sale, etc. Reports available both for accounting and decision making. Requires CBASIC-2. Supplied in source ... **$590/$35**

☐ **JOB COSTING** — Designed for general contractors. Ⓛ To be used interactively with other GRAHAM-DORIAN Ⓜ accounting packages for tracking and analysing ex- †† penses. User establishes customized cost categories and job phases. Permits comparison of actual versus estimated costs. Automatically updates GRAHAM-DORIAN general ledger or runs as stand alone system. Requires CBASIC-2. Supplied in source **$995/$35**

☐ **APARTMENT MANAGEMENT SYSTEM** — Financial Ⓛ management system for receipts and security de- Ⓜ posits of apartment projects. Captures data on va- †† cancies, revenues, etc. for annual trend analysis. Daily report shows late rents, vacancy notices, vacancies, income lost through vacancies, etc. Requires CBASIC-2. Supplied in source **$590/$35**

☐ **CASH REGISTER** — Maintains files on daily sales. Ⓛ Files data by sales person and item. Tracks sales, Ⓜ over-rings, refunds, payouts and total net deposits. †† Requires CBASIC-2. Supplied in source ... **$590/$35**

☐ **tiny C** — Interactive interpretive system for teaching Ⓧ structured programming techniques. Manual includes full source listings **$75/$40**

☐ **BDS C COMPILER** — Supports most major features of Ⓜ language including Structures, Arrays, Pointers, re- Ⓘ cursive function evaluation, linking loader and li- Ⓧ brary. Floating point function library included. Lacks data initialization and static and register class specifiers. Documentation includes "THE C PROGRAMMING LANGUAGE" by Kernighan and Ritchie. **$125/$15**

New Features

☐ **WHITESMITHS C COMPILER** — The ultimate in sys- Ⓛ tems software tools. Produces faster code than Pas- Ⓧ cal with more extensive facilities. Conforms to the full UNIX*** Version 7 C language, described by Kernighan and Ritchie, and makes available over 75 functions for performing I/O, string manipulation and storage allocation. Linkable to Microsoft REL files. Requires 60K CP/M **$630/$30**

☐ **POLYVUE/80** — Full screen editor for any CRT with Ⓜ XY cursor positioning. Includes vertical and horizontal scrolling, interactive search and replace, automatic text wrap around for word processing, operations for manipulating blocks of text, and comprehensive 70 page manual **$135/$15**

☐ **POLYTEXT/80** — Text formatter for word processing Ⓜ applications. Justifies and paginates source text files. Will generate form letters with custom fields and conditional processing. Support for Daisy Wheel printers includes variable pitch justification and motion optimization **$85/$15**

☐ **ALGOL-60** — Powerful block-structured language com- piler featuring economical run time dynamic allocation of memory. Very compact (24K total RAM) system implementing almost all Algol 60 report features plus many powerful extensions including string handling direct disk address I/O etc. Requires Z80 CPU **$199/$20**

☐ **Z80 DEVELOPMENT PACKAGE** — Consists of: (1) disk Ⓜ file line editor, with global inter and intra-line facilities; (2) Z80 relocating assembler, Zilog/Mostek mnemonics, conditional assembly and cross reference table capabilities; (3) linking loader producing absolute Intel hex disk file **$95/$20**

*CP/M is a trademark of Digital Research.
**Z80 is a trademark of Zilog, Inc.
***UNIX is a trademark of Bell Laboratories.
****WHATSIT? is a trademark of Computer Headware.
*****Electric Pencil is a trademark of Michael Shrayer Software.
★TRS-80 is a trademark of Tandy Corp.

†CP/M for Heath, TRS-80 Model I and PolyMorphic 8813 are modified and must use specially compiled versions of system and applications software.
††Recommended system configuration consists of 48K CP/M, 2 full size disk drives, 24 x 80 CRT and 132 column printer.

Ⓜ Modified version available for use with CP/M as implemented on Heath and TRS-80 Model I computers.

Ⓛ User license agreement for this product must be signed and returned to Lifeboat Associates before shipment may be made.

Ⓘ Ⓧ This product includes/eXcludes the language manual recommended in Sundries and Notions above.

☐ **ZDT** — Z80 Monitor Debugger to break and examine Ⓜ registers with standard Zilog/Mostek mnemonic disassembly displays. $35 when ordered with Z80 Development Package **$50/$10**

☐ **DISTEL** — Disk based disassembler to Intel 8080 or TDL/Xitan Z80 source code, listing and cross reference files, Intel or TDL/Xitan pseudo ops optional. Runs on 8080 **$65/$10**

☐ **DISILOG** — As DISTEL to Zilog/Mostek mnemonic Ⓜ files. Runs on Z80 only **$65/$10**

☐ **XASM-68** — Non-macro cross-assembler with nested conditionals and full range of pseudo operations. As sembles from standard Motorola MC6800 mnemonics to Intel hex **$200/$25**

☐ **XASM-65** — As XASM-68 for MOS Technology MCS-6500 series mnemonics **$200/$25**

☐ **TEXTWRITER III** — Text formatter to justify and pagi- Ⓜ nate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil***** prepared files **$125/$20**

☐ **POSTMASTER** — A comprehensive package for mail Ⓜ list maintenance that is completely menu driven. Features include keyed record extraction and label production. A form letter program is included which provides neat letters on single sheet or continuous forms. Compatible with NAD files. Requires CBASIC-2 **$150/$15**

☐ **WHATSIT?**** Interactive data-base system using associative tags to retrieve information by subject. Hashing and random access used for fast response. Requires CBASIC-2 **$125/$25**

☐ **XYBASIC** Interactive Process Control BASIC — Full disk BASIC features plus unique commands to handle bytes, rotate and shift, and to test and set bits. Available in Integer, Extended and ROMable versions. Integer Disk or Integer ROMable **$295/$25**
Extended Disk or Extended ROMable **$395/$25**

☐ **SMAL/80** Structured Macro Assembled Language — Package of powerful general purpose text macro processor and SMAL structured language compiler. SMAL is an assembler language with IF-THEN-ELSE, LOOP-REPEAT-WHILE, DO-END, BEGIN-END constructs **$75/$15**

☐ **SELECTOR III-C2** — Data Base Processor to create †† and maintain multi Key data bases. Prints formatted Ⓜ sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source ... **$295/$50**

☐ **GLECTOR** — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 52K system **$250/$25**

☐ **CPM/374X** — Has full range of functions to create or re-name an IBM 3741 volume, display directory information and edit the data set contents. Provides full file transfer facilities between 3741 volume data sets and CP/M files **$195/$10**

☐ **BASIC UTILITY DISK** — Consists of: (1) CRUNCH-14 Ⓜ — Compacting utility to reduce the size and increase the speed of programs in Microsoft BASIC and TRS-80 BASIC. (2) DPFUN — Double precision subroutines for computing nineteen transcendental functions including square root, natural log, log base 10, sin, arc sin, hyperbolic sin, hyperbolic arc sin, etc. Furnished in source on diskette and documentation ... **$50/$35**

☐ **THE STRING BIT** — FORTRAN character string han- Ⓜ dling. Routines to find, fill, pack, move, separate, concatenate and compare character strings. This package completely eliminates the problems associated with character string handling in FORTRAN. Supplied with source **$45/$15**

☐ **BSTAM** — Utility to link one computer to another also Ⓜ equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansions to send *.COM, etc. 9600 baud with wire, 300 baud with phone connection. **Both** ends need one. Standard and Ⓜ versions can talk to one another. Compatible TRSDOS version also available **$150/$5**

★ ★ ★ ★ ★ ★

SUNDRIES & NOTIONS

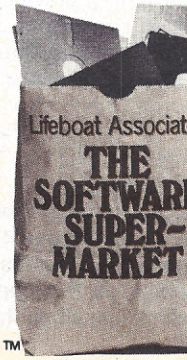
☐ **HEAD CLEANING DISKETTE** — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5" or 8" **$20 ea./$45 for 3**

☐ **FLIPPY DISK KIT** — Template and instructions to modify single sided 5¼" diskettes for use of second side in single sided drives **$12.50**

☐ **FLOPPY SAVER** Protection for center holes of 5¼" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool, tough 7-mil mylar reinforcing rings. Installation tools and rings for 25 diskettes **$14.95**
Re-orders of rings only **$7.95**

☐ **PASCAL USER MANUAL AND REPORT** — By Jensen and Wirth. The standard textbook on the language. Recommended for use by Pascal/Z, Pascal/M and Pascal/MT users **$8**

☐ **THE C PROGRAMMING LANGUAGE** — By Kernighan and Ritchie. The standard textbook on the language. Recommended for use by BDS C, tiny C, and Whitesmiths C users **$12**



Orders must specify disk systems and formats: e.g. North Star single, double or quad density, IBM single or 2D/256, Altair, Helios II, Micropolis Mod I or II, 5¼" soft sector (Micro iCOM/SD Systems Dynabyte), etc.

Prices F.O.B. New York. Shipping, handling and C.O.D. charges extra.

Manual cost applicable against price of subsequent software purchase.

The sale of each proprietary software package conveys a license for use on one system only.

TM

Lifeboat Associates,
2248 Broadway, N.Y., N.Y. 10024

Shopping List No.10

Everything on Shopping List #10 runs on 64K TRS-80 Model II

Software for most popular 8080/Z80 computer disk systems including
NORTH STAR, iCOM, MICROPOLIS, DYNABYTE DB8/2 & DB8/4, EXIDY SORCERER, SD SYSTEMS, ALTAIR, VECTOR MZ, MECA, 8" IBM, HEATH H17 & H89, HELIOS, IMSAI VDP42 & 44, REX, NYLAC, INTERTEC, VISTA V80 and V200, TRS-80 MODEL I and MODEL II,

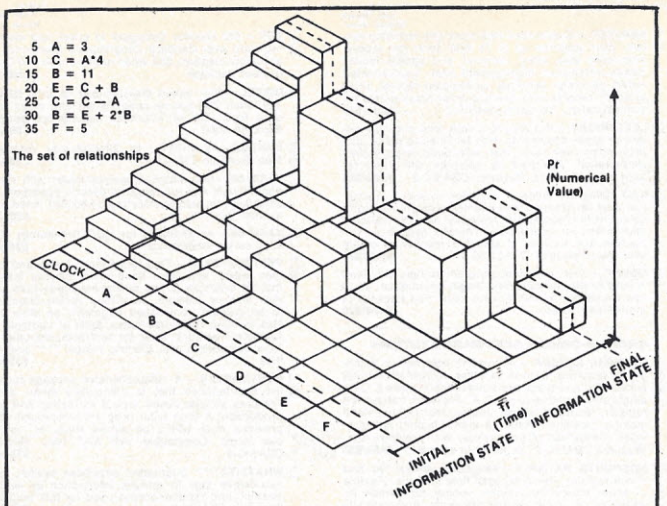


Figure 7. Graphic Representation of the Program

mation states actually occurs. So by our definition it is thinking. Since the program (data and relationships) can be represented by the internal hardware of a computer, when that computer executes the program (*any* program) it is thinking (albeit at a very low level of clustering).

Now, have I been too simplistic in my definition of thinking? Do I really believe that even the execution of a nonsense program like the one above is actually thinking? Haven't I presupposed that computers can think and then tried to find a suitable definition to fit my assumption, which was exactly what I wanted to avoid?

No, yes, and, no. I have attempted to define the verb "to think" with as few presuppositions as possible. Probably *the* most basic assumption which I *have* made is that the ultimate goal of all science is the discovery of consistencies. This, in turn, is based on the assumption that there exists a set of consistencies (eternal truths, if you will).

A corollary to this is the assumption that all consistencies are, by nature, *simple,* so that scientists always opt for the explanation (some call it natural law) which contains the fewest assumptions and is the simplest. The definition of thinking as the transition between information states is the simplest and most concise that I have been able to formulate (which of course does not preclude the formulation of other equally simple definitions).

Perhaps this is best understood when you realize that I have defined *thinking,* not *human thinking.* At the beginning of this article I stated that I would not assume that thinking is inherently and exclusively associated with human beings. It is a phenomenon and I have attempted to define it as such.

An extension of the original question now arises, however. "Can computers think *as humans think*?" That is what is really asked when people want to know if computers can think. Such a question actually has to be divided into two parts. First, *do* computers think as humans think, and secondly, *can* computers think as humans think?

Relying upon my definition of thinking, I feel it is clear that computers *do not* think like you and me. What I mean by this is that no computer *today* thinks at the level of complexity (level of clustering) at which the human brain thinks. It is even clearer that computers do not think in the same *form* (material and structure) as human brains.

The other part of the question is more difficult to answer. "*Can* computers think like humans?" is actually asking "Is it possible to embody the ability to think like humans in a non-human entity?" or "Will machines *ever* be able to think like humans?"

The answer to this cannot be based upon our current level of technological development. It seeks to determine whether thinking at the human level is somehow the exclusive realm of human beings. Is this particular organic entity the only such entity which can attain this level of complexity, or can it be accomplished by some other combination of matter and energy (a computer, for example).

Let's look back at our graphic representation of thinking in Figure 6. We assume some set of elementary data particles and a set of relationships among them, defining their relative positions within the two dimensions. Given the right set of particles and relationships, we will observe clustering, an increase in the complexity of the system.

But perhaps not all particles will cluster. Some will cluster to one degree of complexity and others will cluster to a much higher level. The degree and distribution of clustering is fully defined by the set of elementary data particles and the set of relationships. The question reduces to this, "Does the set of elementary data particles and relationships which define our real universe provide for a variety of high level clustering (human-like thinking) or is only some small subset of all particles capable of clustering to this level?"

Can electricity and semiconductors (or some other physical entities) achieve high level clustering or is it the sole realm of certain organic (read that "carbon") compounds? A very difficult question to answer. We cannot rule out the possibility that compounds of carbon in conjunction with certain other chemicals are the only compounds that can achieve the level of complexity which we commonly call thinking. It may very well be that it is physically impossible to achieve that level with any other compound.

On the other hand, we have already seen that computers of today can perform sophisticated behaviors normally associated with human beings. It is clear that computers can "think" to an impressive degree of complexity. We have yet to encounter a boundary which limits the theoretical complexity of computer programs. It appears, at this stage in our understanding of computers, that they will someday surpass human beings in its complexity, that computers will far exceed us in our ability to think.

The final answer is this; we should remain as agnostics, believing that computers may someday think as we do, even perhaps surpassing us, but recognizing that it may be impossible for them to do so. We can never say, with utter assurance, that they cannot. We can only say that they might.□

UNIVERSE AT LARGE
Continued from Page 90

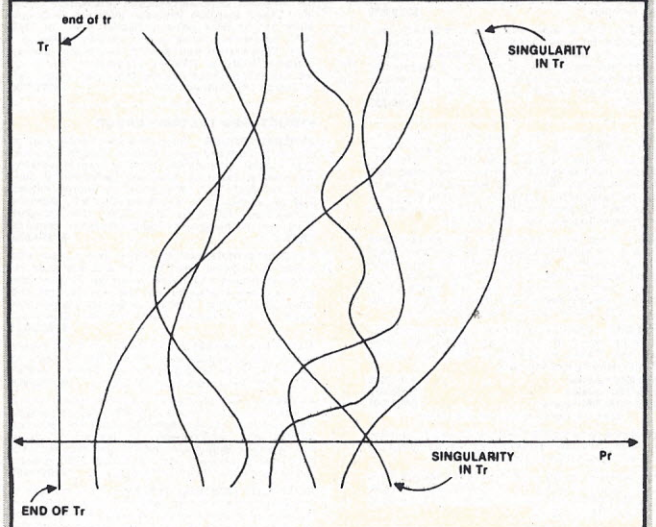


Figure 3a. Double-Ended (finite in Tr) World

While there may be no way that we can determine which type of world our physical universe really is (or even whether this method of modelling the universe comes close to reality) it is interesting to speculate that there might actually be a beginning and end to what we conceptually view as space/time.□

MicroNET

It's off and running. And delivering as promised.

What is MicroNET?

It is the personal computing service of CompuServe, Incorporated. CompuServe is a nationwide commercial time sharing computer network with large-scale mainframes. MicroNET allows the personal computer user access to CompuServe's large computers, software and disc storage during off-peak hours (from 6 PM to 5 AM weekdays, all day on Saturdays, Sundays and most holidays).

What do I get?

You can use our powerful processors with X-Basic, Fortran, Pascal, Macro-10, AID or APL. You get 128K bytes of storage free (just access it at least once a month). Software includes games—including networking multi-player games—personal, business and educational programs.

In addition, there is the MicroNET National Bulletin Board for community affairs, for sale and wanted notices and the MicroNET Electronic Mail System for personal messages to other MicroNET users. You can even sell software via MicroNET.

NEW! MicroQUOTE, a security information system for corporate stocks and public debt.

NEW! MicroNET Software Exchange with dozens of new programs available for downloading to your personal computer at a specified charge.

NEW! Executive programs for TRS-80, Apple II and CP/M systems (so your machine and ours can talk to each other error-free). You can switch between terminal and local mode while on line.

What do I have to have to use MicroNET?

The standard 300 baud modem. MicroNET has local phone service in most major cities (see below) and a reduced phone charge in over a hundred others.

What is the cost?

We've saved the best for last. There is a one-time hook-up charge of only $9.00! Operating time—billed in minutes to your VISA or MasterCharge card—is only $5.00 an hour.

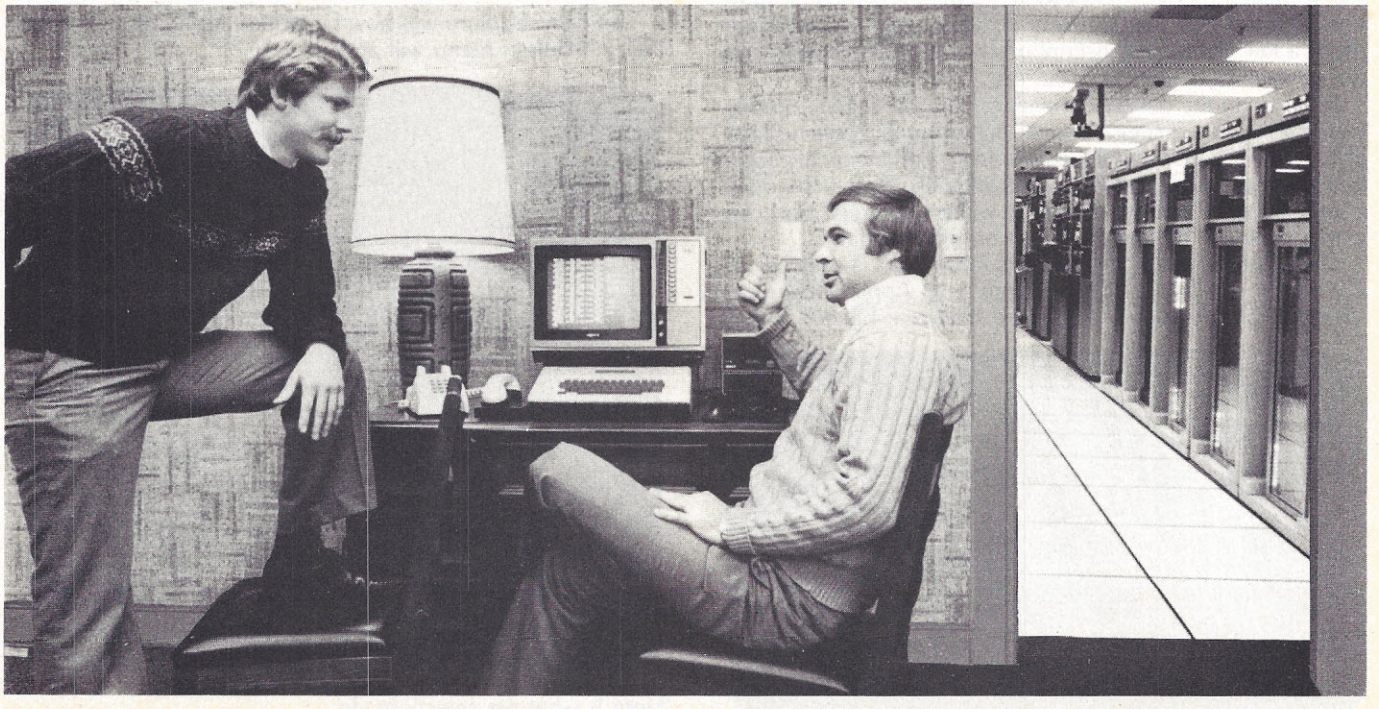
Want more information?

Good. Write to us at the address below. We'll send you a full packet of information about MicroNET.

CompuServe
Personal Computing Division
Dept. I
5000 Arlington Centre Blvd.
Columbus, Ohio 43220

MicroNET is available via local phone calls in the following cities: Akron, Atlanta, Boston, Canton, Chicago, Cincinnati, Cleveland, Columbus, Dallas, Dayton, Denver, Detroit, Houston, Indianapolis, Los Angeles, Louisville, Memphis, West Caldwell (NJ), New York, Philadelphia, Pittsburgh, San Francisco, Stamford (CT), St. Louis, Toledo, Tucson and Washington, D.C.

Access to the MicroNET service is available in 153 other cities for an additional charge of $4.00 per hour.



". . . but the really impressive stuff is in the back room."

Data Base Management for the Apple II

By Phil Roybal
Apple Computer, Inc.

What's a personal computer really good for? Well, some say that computers are for programming. Unfortunately, that may fail to justify the marvelous machines to your neighbors or spouse. But here is a database management program, an application that has immediate appeal to users ranging from business managers to housekeepers.

The management of data is something you do every day. At home, you probably keep a database on friends and family: an address book. In it, information is kept alphabetically by last name, keeping vital information such as friends' addresses, phone numbers, etc. You might also record birthdates, kids' names, clothing sizes, and so on.

But aren't there days when that data is wasted, because you can't find the item you need? I.e., you know that some relative has a birthday this month. Who? Well, you could start at the A's and flip through your address book, looking at each entry. But this approach will take you a while, because you haven't organized your data for that sort of access.

Wouldn't it be nice to say to a computer (as one often can to a grandmother) "Who in the family has a birthday in June?", and get a quick response?

Here in a nutshell is the classic problem. How can you find data under one reference system when it was organized under another? That's where the database manager comes in.

What else can it do? Maybe you could use a little help at the office in the preparation of a new price list. I used to create a new one by marking up an old price list and asking my secretary to retype it. However, typing columns of numbers is not a task that humans do very well, so she might make a few errors in transcription. I would catch some, and the customers would catch the rest.

Now, using the database manager, this job is done quickly and reliably. New entries are sorted according to model number and type of product, and the data is neatly printed out in a form that closely resembles the finished price list. Best of all, once I have correctly entered a price, it will stay correct. So I can concentrate on the new data, not on proofreading.

The database manager that does these tasks is called FILE CABINET. It is available through any Apple dealer on Volume 3 of Apple's Contributed Software Library. It will help you to retrieve, reorganize, and report upon data in ways that you probably wouldn't or couldn't do by hand.

```
NAME FOR NEW DATA BASE FILE: POSSESSIONS
```

PROGRAM OPERATION

FILE CABINET is easy to run. When using it for the first time, you'll be asked to name the database to be created. If it is to be a list of any valuables with serial numbers, retail value, etc., it might be called "POSSESSIONS".

After naming the file, its structure must be defined: the categories under which data will be stored. In this database, a typical set of headings might be entered like this:

```
HEADING FOR COLUMN NUMBER 1: ITEM

HEADING FOR COLUMN NUMBER 2: PURCHASED

HEADING FOR COLUMN NUMBER 3: COST

HEADING FOR COLUMN NUMBER 4: MODEL #

HEADING FOR COLUMN NUMBER 5: SERIAL #

HEADING FOR COLUMN NUMBER 6: <press
RETURN to end this stage of
the program>
```

The data structure will now be written to the diskette, and then a 10-item menu will appear.

```
1. SELECT DATA BASE

2. SEARCH AND/OR CHANGE DATA

3. ADD RECORDS

4. DELETE RECORDS

5. REPORT

6. SORT (TAKES APPROX. n MINUTES)

7. TURN ON PRINTER

8. TURN OFF PRINTER

9. LIST DATA BASE

10. QUIT
```

Here is what the selections do.

Select Data Base

You actually started here when you "booted" up the disk. FILE CABINET allows you to select the database you want: lists of friends, creditors, possessions, etc. If there are no databases on the disk, as there weren't when we first used the program, you are invited to create one. Any time later, you can also exercise this option.

Search And/Or Change Data

This very useful section of FILE CABINET lets the computer's fingers do the walking through the database; searching under any heading you choose for a particular word, letter, or number. When it finds it (if it does), the entire record is displayed on the screen.

The same section of the program lets you change a record that you previously entered. Just specify the record number to be altered and the complete record will be displayed. Select the field you want to change and type in the correct information. The record will be corrected on the screen. When you have made all the changes you want, the program will write the corrected database back out to disk.

Add Records

Databases without data aren't much use, so this is an important section. When you enter it, the computer will display the number of records stored already, tell you the record number you're entering, and ask you what is to be stored under each of the headers you previously defined. Enter records until you've had enough. Then the computer will save them for you on disk.

Delete Records

Here you can get rid of an entire record at once. Just specify the record number. Subsequent records will be renumbered to close the gap.

Report

There's nothing like seeing your labors rewarded with a neatly printed report. This is the command which does it for you.

To start with, the system asks you to define a report format. Decide what headings from the database will be included, where they will be positioned in the report, and whether they will be totalled or not. If you request column totals, the system will ask you the position for a horizontal total column. If you just respond with RETURN, no horizontal totals will be printed.

Once you have said what your report looks like, you tell the computer how to decide whether a record should appear in it. Of course, you may just want to print everything. But if you would like to be selective (perhaps to get a list of only those friends who sent you Christmas cards last year), here's your chance. The computer will ask:



To show the entire database, just press RETURN. To select a subset of the data, enter the number of the column the computer should scan to select records. If you requested a selection based upon a header called "CHRISTMAS" for instance, the system might then prompt you:



By entering a "key" for the computer to look for (date of the last Christmas card in this example), you can have only those records in which the key is found printed in your report.

Don't be discouraged if your first report doesn't look quite as grand as you had imagined it. It takes a little practice to get the best results, just as it does with any other tool.

Sort

Suppose you're preparing a report for the insurance company after a disaster in which everything but your faithful Apple was destroyed. You may want all the items that were in each room to appear together. Or you may want them to be printed out in order of value. SORT lets you rearrange records alphabetically or numerically based upon the contents of any heading in the data structure.

Turn On Printer

This menu option allows you to obtain printed output instead of (or in addition to) a display on the screen. When you select it, the program will ask you the line width to be printed. If you select 40 columns, your report will be displayed and printed. For greater line widths, the screen display will be disabled while the report is printed. The program is designed for printers interfaced through slot #1 of the Apple.

Turn Off Printer

This command disables the printer.

List

This menu option will display or print the entire contents of the database.

Quit

QUIT puts you back into Applesoft BASIC.

DATA STRUCTURE AND FILES

FILE CABINET finds its way around using tables of information that it stores on the disk. Let's embark upon a brief tour of these tables.

BASENAMEFILE is created when this program is run for the first time. It is a list of all the databases stored on the disk. Figure 1 shows its format. Each time you add a database, this file gets rewritten.

```
RECORD 0        NO. OF DATABASES ON FILE
RECORD 1        NAME OF FIRST  DATA BASE




RECORD N        NAME OF LAST DATA BASE
```

Figure 1. BASENAMEFILE Format

All the rest of the files are particular to a given database, and the name of that database is the first part of the file name. If your database was called "POSSESSIONS" then the remaining files would be:

POSSESSIONS HEADERFILE — When you created your database, you gave it structure by describing what headings data would be organized under. That heading information is stored in this file, as shown in Figure 2.

```
RECORD 0        NO. OF HEADERS IN DATA STRUCTURE
RECORD 1        FIRST HEADER
RECORD 2        SECOND HEADER
   .
   .
   .
RECORD N        LAST HEADER
```

Figure 2. HEADERFILE Format

POSSESSIONS INDEXFILE — This is the big one; the actual database. Records are stored sequentially in it as ASCII strings. See Figure 3 for the details.

```
RECORD 0        NO. OF RECORDS IN THIS DATA BASE
RECORD 1        FIRST ENTRY OF FIRST  RECORD
RECORD 2        SECOND ENTRY OF  FIRST RECORD
   .
   .
   .
RECORD M        LAST ENTRY OF FIRST RECORD
RECORD M+1      FIRST ENTRY OF  SECOND RECORD
RECORD M+2      SECOND ENTRY OF  SECOND RECORD
   .
   .
   .
RECORD N        LAST ENTRY OF LAST RECORD
```

Figure 3. INDEXFILE Format

POSSESSIONS RPTFMTNAMEFILE — This file with the $10 name lists all the report formats you defined for a particular database. Note that this file is connected by its name to the POSSESSIONS database. Thus it cannot be used with a different database like "FRIENDS", even though report formats may be identical. See Figure 4 for its structure.

```
RECORD 0        NO. OF REPORT FORMATS DEFINED
RECORD 1        NAME OF FIRST FORMAT
RECORD 2        NAME OF SECOND FORMAT
   .
   .
   .
RECORD N        NAME OF LAST FORMAT
```

Figure 4. RPTFMTNAMEFILE

POSSESSIONS RPTFMTFILE — This file stores each report format referred to by the name in the above file. See Figure 5.

```
RECORD 0        NO. OF RECORDS IN  RPTFMTFILE FILE
RECORD 1        NO. OF FIRST HEADER IN REPORT
RECORD 2        TAB POS'N OF FIRST HEADER COLUMN
RECORD 3        0=NO TOTAL, 1=TOTAL
RECORD 4        NO. OF SECOND HEADER IN REPORT
RECORD 5        TAB POS'N OF SECOND HEADER COLUMN
RECORD 6        0=NO TOTAL, 1=TOTAL
   .
   .
   .
RECORD N-2      NO. OF LAST HEADER IN REPORT
RECORD N-1      TAB POS'N OF LAST HEADER COLUMN
RECORD N        0=NO TOTAL, 1=TOTAL
RECORD N+1      0=NO HORIZ. TOTALS, 1=HORIZ. TOTALS
RECORD N+2      TAB POS'N OF HORIZ. TOTAL COLUMN
```

Figure 5. RPTFMTFILE

It's handy to know what the files are when you want to do things that the program was not designed to do — such as alter the names of headings in a database that has already been created. By writing a small program that OPENs the POSSESSIONS HEADERFILE and rewrites it with different names (be sure not to change the number of names), you can make the changes you need. This is useful when you are maintaining a price list and your price break quantities (column headings) change.

THE CODE

Listing 1 shows the Applesoft BASIC code that the program is written in. It runs in any 32K or larger Apple II Plus (or Apple II with Applesoft BASIC). It requires a floppy disk and optionally may use a printer.

The code is modular, with well-defined sections that handle different tasks. Each major section begins with

REM ***section name***

Thus, if you were going to change the REPORT section, you would find it starting at line 2940 with REM ***REPORT***. I have had the most occasion to make changes to this section, and you may too. For instance, the program prints reports with the headings distributed across the page. If you want to print mailing labels, you would have to change it so that all information was printed at the left margin with a line feed after each entry.

The REPORT generator starts by storing the header number to be scanned in a variable named S and the criterion in string Q$. It then goes through the data a record at a time for a match to Q$ in heading number S. (See code lines 3260 to 3350).

If a match is found, the computer jumps down to line 3440, picks up the data from the headings you included in your report, tabs over to the appropriate column on the printer or display, and prints the datum (see lines 3440 to 3470). When it has printed one entire record, it issues a line feed. The program continues on through the database until the record counter, J, is equal to the length of the database, NR.

I've found FILE CABINET to be one of the most useful programs available for my Apple; and I have had no trouble in adapting it to a variety of applications. Give it a try. It'll prove handy the next time a friend asks, "What do you do with your computer?"□

Program follows

Operating Power!

with

I/OS™ Version 3.0

8080/Z80 OPERATING SYSTEM

The former TSA/OS has been significantly upgraded and improved...

- New Spooler
- Hard disk support
- Big files supported (over 200,000,000 bytes)
- New system support functions and utilities
- Improved versions of standard utilities
- New improved manuals

$150 gets it all **I/OS,** spooler, utilities, manuals!

(plus dealer configuration charges)

Now also available for the Radio Shack TRS 80 MODEL II!

A BIG SYSTEM OS FOR YOUR MICRO!

- True 'sysgen' for an optimized system for any hardware
- Support for most popular terminals, devices and systems plus user driver interface
- CP/M* — CDOS* — SDOS* compatible
- Supports systems with multiple disk types (two or more drivers)

*™ Digital Research, Cromemco, SD Systems

SPECIAL OFFER

for

TRS 80 MODEL II — I/OS

PLUS

the Wp Daisy™ Word Processor...

$550

total price!

★★NEXT MONTH★★

I/Pascal™ compiler

I/SAL™ structured assembler

...more to follow!

★ ★ ★ ★ ★

Software for I/OS — TSA/OS — CDOS — SDOS — other 8080/Z80 OS*M operating systems

Contact your dealer or...

InfoSoft SYSTEMS INCORPORATED

25 SYLVAN ROAD SOUTH
WESTPORT, CONN. 06880
(203) 226-8937

PROGRAM LISTING

```
0  POKE 216,0
1000  PRINT "MAXFILES1"
1010 D$ = "": REM CTRL D
1020  PRINT D$"NOMON,I,O,C"
1030  TEXT : HOME
1040  GOSUB 5180
1050  CLEAR
1060  DIM R$(65),AC(21),K(65),H$(21),RN$(21)
1070  DIM Z$(21)
1080 COMMA$ = "NO"
1090 D$ = "": REM CTRL D
1100 H$(0) = "REC#"
1110 DB$ = "":F$ = "BASENAME": ONERR  GOTO 2610
1120  GOSUB 4110
1130  GOTO 2470
1140 F$ = "HEADER": ONERR  GOTO 1520
1150  GOSUB 4110
1160  FOR I = 1 TO NR:H$(I) = R$(I): NEXT I
1170 NH = NR:NR = 0:MEM =  FRE (0)
1180 B =  INT (MEM / (13 * NH))
1190  DIM N$(B,NH),R(B)
1200 F$ = "INDEX": ONERR  GOTO 4830
1210  GOSUB 4110
1220  GOTO 4810
1230  REM *** SORT ***
1240  FOR I = 1 TO NR:R(I) = 0: NEXT I
1250  FOR I = 1 TO NR: FOR J = 1 TO NR
1260  ON L GOTO 1270,1290
1270  IF N$(I,S) =  > N$(J,S) THEN R(I) = R(I) + 1
1280  GOTO 1300
1290  IF  VAL (N$(I,S)) =  >  VAL (N$(J,S)) THEN R(I) = R(I) + 1
1300  NEXT J: NEXT I
1310  PRINT "SORTING ";
1320  FOR I = NR TO 1 STEP  - 1: FOR J = NR TO 1 STEP  - 1
1330  IF I <  > J THEN  IF R(I) = R(J) THEN R(J) = R(J) - 1
1340  NEXT J: NEXT I
1350  PRINT "SORTING "
1360 J = 1
1370  IF R(J) = J THEN J = J + 1: GOTO 1370
1380  IF J >  = NR THEN 1420
1390  FOR I = 1 TO NH:Z$(I) = N$(R(J),I):N$(R(J),I) = N$(J,I):N$(J,I) = Z$
      (I): NEXT I
1400 Z = R(R(J)):R(R(J)) = R(J):R(J) = Z
1410  GOTO 1370
1420  PRINT "": PRINT "WANT TO SAVE THE "DB$" FILE": PRINT "SORTED BY "H$(
      S)" TO DISK ";: INPUT "Y/N) ?";L$: IF L$ = "Y" THEN F$ = "INDEX": GOSUB
      4280
1430  GOTO 4810
1440 MF = 1: GOSUB 3880
1450  INPUT "ENTER # OF FIELD FOR SORT ";S$:S =  VAL (S$): IF S < 1 OR S >
      NH THEN 1450
1460  PRINT : PRINT "DO YOU WANT TO SORT:": PRINT
1470  PRINT "1 ALPHABETICALLY"
2040  IF S < 1 OR S > 3 THEN 2030
2050  ON S GOTO 1800,2080,4810
2060  IF PF <  > 0 THEN 2070
2062  PRINT "PRESS RETURN TO CONTINUE, ESC FOR MENU";
2064  GET L$
2065  IF  ASC (L$) = 27 THEN 4810
2066  IF  ASC (L$) <  > 13 THEN 2064
2070 L = 0: HOME : RETURN
2080  REM ***CHANGE DATA***
2090  PRINT "
ENTER THE NUMBER OF THE RECORD"
2100  INPUT "YOU WANT TO CHANGE ";J$:J =  VAL (J$)
2110  HOME : GOSUB 2240
2120  PRINT "
ENTER THE NUMBER OF THE FIELD YOU WANT": PRINT "TO CHANGE ";
2130  INPUT "";S$:S =  VAL (S$)
2140  IF S < 1 OR S > NH THEN 2130
2150  PRINT
2160  PRINT "FROM "H$(S)": "N$(J,S)
2170  PRINT
2180  PRINT "TO "H$(S)": ";: INPUT "";N$(J,S)
2190  HOME : GOSUB 2240
2200  PRINT
2210  INPUT "
MORE CHANGES (Y/N) ";L$
2220  IF L$ = "Y" THEN 2080
2230 F$ = "INDEX": GOSUB 4280: GOTO 4810
2240  REM ***PRINT A RECORD***
2250  ON PF GOSUB 5230,5250
2260  PRINT "  "H$(0)": ";J
2270  FOR I = 1 TO NH
2280  PRINT I" "H$(I)": "N$(J,I)
2290  NEXT I
2300  PRINT
2310 L = L + NH + 2
2320  PRINT D$"PR#0"
2330  RETURN
2340  REM ***DELETE RECORDS***
2350  HOME
2360  INPUT "ENTER RECORD NUMBER YOU WANT DELETED ";DR$:DR =  VAL (DR$)
2370  IF DR < 1 OR DR > NR THEN 2360
2380  FOR J = DR TO NR - 1
2390  FOR I = 1 TO NH
2400 N$(J,I) = N$(J + 1,I)
2410  NEXT I
2420  NEXT J
2430  PRINT : PRINT "RECORD NUMBER "DR" DELETED!": PRINT
2440  INPUT "MORE (Y/N) ";L$
2450  IF L$ = "Y" THEN 2360
2460 NR = NR - 1:F$ = "INDEX": GOSUB 4280: GOTO 4810
2470  REM *** BASENAMEFILE ROUTINES ***
2480  HOME
2490  PRINT "SELECT FROM:": PRINT
2500  FOR J = 1 TO NR: PRINT J" "R$(J): NEXT J: PRINT
2510  PRINT J" CREATE A NEW DATA BASE"
2520  IF J > 1 THEN  PRINT J + 1" DELETE A DATA BASE"
2530  PRINT
2540  INPUT "WHICH ";S$:S =  VAL (S$)
```

```
1480  PRINT "2 NUMERICALLY"
1490  PRINT
1500  INPUT "WHICH ";L$:L =  VAL (L$)
1510  PRINT : PRINT "SORTING ";: GOTO 1240
1520  CALL 1013: REM *** CREATE HEADERFILE ***
1530 NR = 1
1540  HOME : PRINT "PRESS 'RETURN' TO EXIT TO MENU"
1550  PRINT
1560  PRINT "HEADER FOR COLUMN NUMBER "NR": ";: INPUT "";R$(NR)
1570  IF R$(NR) = "" OR NR > 20 THEN 1600
1580 NR = NR + 1
1590  GOTO 1560
1600 NR = NR - 1
1610  GOSUB 4280: GOTO 1160
1620  REM ***ENTER RECORDS***
1630  HOME
1640  PRINT "THERE ARE "NR" RECORDS"
1650  PRINT "IN THE "DB$" FILE"
1660 NR = NR + 1
1670  PRINT "YOU ARE ENTERING RECORD # "NR
1680  PRINT
1690  FOR I = 1 TO NH
1700  PRINT H$(I)":";: GOSUB 4720:N$(NR,I) = I$
1710  NEXT I
1720  PRINT
1730  INPUT "MORE (Y/N) ";L$
1740  IF L$ = "Y" THEN 1640
1750 F$ = "INDEX"
1760  GOSUB 4280
1770  GOTO 4810
1780  REM ***SEARCH/CHANGE***
1790 L = 0
1800  HOME
1810  PRINT "YOU MAY SEARCH BY ANY OF THE FOLLOWING:"
1820  PRINT
1830  GOSUB 3880
1840  PRINT : PRINT "OR YOU MAY": PRINT
1850  PRINT I" MAKE CHANGES"
1860  PRINT
1870  INPUT "WHICH ";S$:S =  VAL (S$)
1880  IF S < 0 OR S > NH + 1 THEN 1870
1890  IF S = NH + 1 THEN 2080
1900  HOME
1910  PRINT "PLEASE ENTER THE "H$(S): PRINT "YOU WANT TO FIND.....
": INPUT
     "";Q$
1920  HOME
1930  FOR J = 1 TO NR
1940 N$(J,0) =  STR$ (J)
1950  IF  LEFT$ (N$(J,S), LEN (Q$)) = Q$ THEN  GOSUB 2240
1960  IF L + NH > 20 THEN  GOSUB 2060
1970  NEXT J
1980  PRINT "THAT'S ALL OF THEM. ";
1990  PRINT "NOW YOU MAY:"
2000  PRINT "1 DO MORE SEARCHES"
2010  PRINT "2 MAKE CHANGES"
2020  PRINT "3 RETURN TO THE MAIN MENU"
2030  INPUT "
WHICH ";S$:S =  VAL (S$)
2550  IF S = J + 1 THEN 2660
2560  IF S < 1 OR S > J THEN  PRINT  CHR$ (7);: VTAB  PEEK (37): CALL  - 8
     68: GOTO 2540
2570 DB$ = R$(S)
2580  IF S <  > J THEN 1140
2590  PRINT
2600  GOTO 2620
2610  CALL 1013

2620  IF J = 0 THEN J = 1
2630  INPUT "NAME FOR NEW DATA BASE FILE :";R$(J)
2640 NR = J: GOSUB 4280
2650 DB$ = R$(J - 1): GOTO 1140
2660  REM      *** DELETE A DATA BASE ***
2670  PRINT : INPUT "DELETE WHICH : ";S$:S =  VAL (S$)
2680  IF S < 1 OR S > J - 1 THEN  PRINT  CHR$ (7);: VTAB  PEEK (37) - 1:
 CALL - 868: GOTO 2670
2690  HOME : VTAB (9): PRINT "READY TO DELETE " CHR$ (34);R$(S); CHR$ (34)
     ;".": PRINT
2700  PRINT "ONCE DELETED, THIS DATA CANNOT BE"
2710  PRINT "RECOVERED.  ARE YOU SURE THAT YOU"
2720  PRINT "WANT TO DELETE IT? (Y/N) ";: INPUT "";S$
2730  IF S$ <  > "Y" THEN 2470
2740 HOME : VTAB 12: HTAB 11: INVERSE : PRINT  CHR$ (91);" DELETING DATAB
     ASE ]": NORMAL
2750  ONERR  GOTO 2830
2760 DB$ = R$(S)
2770 F$ = "RPTFMTNAME"
2780  GOSUB 4110
2790  PRINT D$;"DELETE"DB$" "F$"FILE"
2800  FOR I = 1 TO NR
2810  PRINT D$;"DELETE"DB$" "R$(I)" RPTFMTFILE"
2820  NEXT I
2830  CALL 1013: PRINT D$"DELETE"DB$" RPTFMTNAMEFILE"
2840  PRINT D$"DELETE"DB$" INDEXFILE"
2850  PRINT D$"DELETE"DB$" HEADERFILE"
2860 DB$ = ""
2870 F$ = "BASENAME": GOSUB 4110
2880  IF NR = 1 THEN  PRINT D$"DELETE BASENAMEFILE": GOTO 1000
2890  FOR I = S TO NR - 1
2900 R$(I) = R$(I + 1)
2910  NEXT I
2920 NR = NR - 1: GOSUB 4280
2930  GOTO 2470
2940  REM ***REPORT***
2950 T9 = 0
2960  HOME : E = 0
2970  FOR I = 0 TO 3 * NH + 2:K(I) = 0: NEXT I
2980  FOR I = 0 TO NH:AC(I) = 0: NEXT I:HC = 0:GT = 0
2990  ON E GOTO 3150
3000  GOTO 3940
3010  PRINT : INPUT "HOW MANY HEADERS ";RH$:RH =  VAL (RH$): IF RH < 1 OR
     RH > NH + 1 THEN 3010
3020  IF E = 0 THEN RN$(NN) = "PRESENT"
3030  FOR I = 1 TO RH * 3 STEP 3
3040  PRINT "ENTER # OF HEADER YOU WANT IN": PRINT "POSITION #"(I + 2) / 3
     " ";: INPUT "";K$;K(I) =  VAL (K$)
3050  IF K(I) < 0 OR K(I) > NH THEN 3040
3060  PRINT "ENTER TAB FOR "H$(K(I))" ";: INPUT "";K$:K(I + 1) =  VAL (K$)
```

```
3070  IF K(I + 1) < 0 OR K(I + 1) > 255 THEN 3060
3080  PRINT "TOTAL ON "H$(K(I))" (Y/N) ";: INPUT L$
3090  IF L$ = "Y" THEN K(I + 2) = 1:K(0) = 1
3100  NEXT I
3110  IF K(0) <  > 1 THEN 3150
3120  INPUT "ENTER TAB FOR TOTAL: ";A$
3130  IF  LEN (A$) = 0 THEN K(0) = 0:T9 = 1: GOTO 3150
3140 K(I + 1) =  VAL (A$): IF K(I + 1) < 0 OR K(I + 1) > 131 THEN  PRINT "
     ": VTAB  PEEK (37) - 1: GOTO 3120

3150  PRINT
3160  INPUT "SELECT RECORDS BY WHICH HEADER # ";S$:S =  VAL (S$)
3170  IF  LEN (S$) = 0 THEN Q$ = "@": GOTO 3230
3180  PRINT : INPUT "'AND' 2ND HEADER (Y/N) ";L$: IF L$ <  > "Y" THEN X$ =
     "@": GOTO 3200
3190  PRINT : INPUT "ENTER # OF 'AND' HEADER ";X$:X =  VAL (X$)
3200  PRINT : PRINT "@ WILL SELECT ALL RECORDS."
3210  PRINT : PRINT "SELECT RECORDS FOR "H$(S)"= ";: INPUT "";Q$: PRINT
3220  IF L$ = "Y" THEN  PRINT "AND "H$(X)"= ";: INPUT "";X$
3230  FOR I = 1 TO RH + 1: IF K(3 * I - 1) > 35 THEN PF = 2
3240  NEXT I
3250  ON PF GOSUB 5230,5250,5280: GOSUB 3610
3260  FOR J = 1 TO NR
3270 N$(J,0) =  STR$ (J)
3280  IF Q$ = "@" THEN 3320
3290  IF  LEFT$ (N$(J,S), LEN (Q$)) <  > Q$ THEN 3330
3300  IF X$ = "@" THEN 3320
3310  IF  LEFT$ (N$(J,X), LEN (X$)) <  > X$ THEN 3330
3320  GOSUB 3440
3330  IF PF < 1 THEN  IF L > 18 THEN  GOSUB 2060: GOSUB 3610
3340  IF L = 0 THEN  GOSUB 3610
3350  NEXT J
3360  ON T9 GOSUB 3540
3370  PRINT D$"PR#0"
3380  ON E GOTO 3410
3390  PRINT : PRINT "DO YOU WANT TO SAVE THE FORMAT": INPUT "FOR THIS REPO
     RT TO DISK (Y/N) ";L$
3400  IF L$ = "Y" THEN E = 1: GOSUB 3720
3410  PRINT : PRINT "MORE REPORTS USING THE "RN$(NN)" FORMAT": INPUT "(Y/N
     ) ";L$
3420  IF L$ = "Y" THEN  GOSUB 3880:E = 1: GOTO 2980
3430  GOTO 4810
3440  FOR I = 1 TO RH
3450  POKE 36,K(3 * I - 1): PRINT N$(J,K(3 * I - 2));
3460  ON K(3 * I) GOSUB 3510
3470  NEXT I
3480  IF K(0) = 1 THEN  IF HC <  > 0 THEN  POKE 36,K(3 * I - 1): PRINT HC;
     :GT = GT + HC:HC = 0
3490 L = L + 1
3500  PRINT : RETURN
3510 N = 3 * I - 2
3520 V =  VAL (N$(J,K(N))):AC(I) = AC(I) + V:HC = HC + V
3530  RETURN
3540  FOR I = 1 TO 39 + ((PF > 1) * 39): PRINT "-";: NEXT I: PRINT
3550  FOR I = 1 TO RH
3560  IF AC(I) = 0 THEN 3580
3570  POKE 36,K(3 * I - 1): PRINT AC(I);
3580  NEXT I
3590  IF GT <  > 0 THEN  POKE 36,K(3 * I - 1): PRINT GT;
4170  ON FF GOTO 4230
4180  FOR I = 1 TO NH
4190  GOSUB 4720
4200 N$(J,I) = I$
4210  NEXT I
4220  GOTO 4240
4230  INPUT R$(J)
4240  NEXT J
4250  PRINT D$"CLOSE"
4260 FF = 0
4270  RETURN
4280  REM *** SAVE FILES ***
4290  IF F$ <  > "INDEX" THEN FF = 1
4300  PRINT D$"OPEN"DB$" "F$"FILE"
4310  PRINT D$"WRITE"DB$" "F$"FILE"
4320  PRINT NR
4330  FOR J = 1 TO NR
4340  ON FF GOTO 4390
4350  FOR I = 1 TO NH
4360  PRINT N$(J,I)
4370  NEXT I
4380  GOTO 4400
4390  PRINT R$(J)
4400  NEXT J
4410  PRINT D$"CLOSE"
4420 FF = 0
4430  RETURN
4440  REM *** SAVE REPORTNAMEFILE ***
4450 NR = NN
4460 F$ = "RPTFMTNAME"
4470  FOR I = 1 TO NR:R$(I) = RN$(I): NEXT I
4480  GOSUB 4280
4490 NR = NS: RETURN
4500  REM  *** LIST ***
4510 L = 0
4520  HOME
4530  FOR J = 1 TO NR
4540  ON PF GOSUB 5230,5250,5280
4550  PRINT "  "H$(0)": ";J:L = L + 1
4560  FOR I = 1 TO NH
4570  PRINT I" "H$(I)": "N$(J,I)
4580 L = L + 1
4590  NEXT I
4600  PRINT :L = L + 1
4610  IF L + NH > 20 THEN 4660
4620  NEXT J
4630  PRINT D$"PR#0"
4640  INPUT "HIT RETURN FOR MENU...";L$
4650  GOTO 4810
4660  PRINT D$"PR#0"
4670  PRINT "PRESS RETURN TO CONTINUE, ESC FOR MENU";
4672  GET L$
4674  IF  ASC (L$) = 27 THEN 4810
4676  IF  ASC (L$) = 13 THEN 4680
4678  GOTO 4672
4680  HOME :L = 0
4690  ON PF GOSUB 5230,5250,5280
4700  GOTO 4620
4710  STOP
```

```
3600  PRINT : RETURN
3610  HOME
3620  PRINT RN$(NN)" REPORT FOR "H$(S)":"G$;
3630  IF X$ = "@" THEN 3650
3640  PRINT " AND "H$(X)":"X$: GOTO 3660
3650  PRINT "
"
3660  FOR I = 1 TO RH
3670  POKE 36,K(3 * I - 1): PRINT H$(K(3 * I - 2));
3680  NEXT I
3690  IF K(0) = 1 THEN  POKE 36,K(3 * I - 1): PRINT "TOTAL";
3700  PRINT : PRINT
3710 L = 4: RETURN
3720  REM *** SET-UP TO SAVE RPTFMTFILE ***
3730 NS = NR
3740  PRINT : INPUT "ENTER THE REPORT FORMAT NAME ";RN$(NN)
3750 F$ = RN$(NN) + " RPTFMT"
3760 NR = 3 * RH + 2
3770  FOR I = 1 TO NR:R$(I) =  STR$ (K(I)): NEXT I
3780 R$(I - 2) =  STR$ (K(0))
3790  GOSUB 4280: GOSUB 4440
3800  RETURN
3810  REM *** SET-UP TO READ RPTFMTFILE ***
3820 F$ = RN$(NN) + " RPTFMT"
3830  GOSUB 4110
3840 RH = (NR - 2) / 3: FOR I = 1 TO NR:K(I) =  VAL (R$(I)): NEXT I
3850 K(0) =  VAL (R$(I - 2))
3860 NR = NS
3870  GOSUB 3880: PRINT : GOTO 3160
3880  REM *** SUB MENU ***
3890  HOME : PRINT "SELECT FROM:": PRINT
3900  IF MF = 0 THEN  PRINT "0 "H$(0)
3910  FOR I = 1 TO NH: PRINT I" "H$(I): NEXT I: PRINT
3920 MF = 0
3930  RETURN
3940  REM *** READ REPORTNAMEFILE & SELECT REPORT ***
3950 NN = 0: FOR I = 0 TO 21:RN$(I) = "": NEXT I:NS = NR
3960 F$ = "RPTFMTNAME"
3970  ONERR  GOTO 4070
3980  GOSUB 4110
3990  FOR I = 1 TO NR:RN$(I) = R$(I): NEXT I
4000  HOME : PRINT "SELECT FROM:": PRINT
4010  FOR I = 1 TO NR: PRINT I" "R$(I): NEXT I: PRINT
4020  PRINT I" CREATE A NEW REPORT FORMAT": PRINT
4030  INPUT "WHICH ";S$:S =  VAL (S$): IF S < 1 OR S > I THEN 4030
4040 NN = S
4050  IF S <  > I THEN RN$(S) = R$(S):E = 1:NR = NS: GOTO 3810
4060  GOTO 4100
4070  CALL 1013: HOME : PRINT "NO REPORT FORMATS ON DISK...": PRINT
4080 NN = 1
4090  INPUT "DO YOU WANT TO CREATE ONE (Y/N) ?";L$: IF L$ <  > "Y" THEN 48
     10
4100  GOSUB 3880:NR = NS: GOTO 3010
4110  REM *** READ FILES ***
4120  IF F$ <  > "INDEX" THEN FF = 1
4130  PRINT D$"OPEN"DB$" "F$"FILE"
4140  PRINT D$"READ"DB$" "F$"FILE"
4150  INPUT NR
4160  FOR J = 1 TO NR
```

```
4720  REM   ***  INPUT ROUTINES  ***
4730 I$ = ""
4740  IF COMMA$ = "NO" THEN  INPUT "";I$: RETURN
4750  GET A$
4760  IF A$ =  CHR$ (3) THEN  STOP
4770  PRINT A$;
4780  IF A$ =  CHR$ (13) THEN  RETURN
4790 I$ = I$ + A$
4800  GOTO 4750
4810  REM *** MAIN MENU ***
4820  GOTO 4840
4830  CALL 1013
4840  HOME
4850  PRINT "******* DATA BASE MANAGEMENT I *******"
4860  PRINT : PRINT "           APPLE COMPUTER INC"
4870  PRINT
4880  PRINT "CURRENT DATA BASE: "DB$: PRINT
4890  PRINT "CURRENTLY CONTAINS: "NR" RECORDS": PRINT : PRINT "ROOM FOR "B
     - NR" MORE RECORDS"
4900  PRINT
4910  IF PF >  = 1 THEN  PRINT "THE PRINTER IS ";: FLASH : PRINT "ON":NORMAL
     : GOTO 4930
4920  PRINT "THE PRINTER IS OFF"
4930  PRINT
4940  PRINT "1  SELECT DATA BASE"
4950  PRINT "2  SEARCH AND/OR CHANGE DATA"
4960  PRINT "3  ENTER RECORDS"
4970  PRINT "4  DELETE RECORDS"
4980  PRINT "5  REPORT"
4990  PRINT "6  SORT (TAKES APPROX.  " INT (.0005 * NR ^ 2 + .03 * NR)" MIN
     )"
5000  PRINT "7  TURN ON PRINTER"
5010  PRINT "8  TURN OFF PRINTER"
5020  PRINT "9  LIST ALL RECORDS"
5030  PRINT "10 QUIT"
5040  PRINT
5050  INPUT "WHICH ";S$:S =  VAL (S$)
5060  IF S < 1 OR S > 10 THEN 4810
5070  ON S GOTO 1050, 1780, 1620, 2340, 2940, 1440, 5080, 5160, 4500, 5170
5080  HOME
5090  PRINT "PRINTER OPTIONS:"
5100  PRINT "1 40 COLUMNS"
5110  PRINT "2 80 COLUMNS"
5120  PRINT "3 132 COLUMNS"
5130  PRINT : INPUT "WHICH ";PF$:PF =  VAL (PF$)
5140  IF PF < 1 OR PF > 3 THEN 5130
5150  GOTO 4810
5160 PF = 0: GOTO 4810
5170  END
5180  REM *** APPLESOFT ONERR CORRECTION
5190  FOR I = 1013 TO 1022: READ PP: POKE I,PP: NEXT I
5200 I = 0
5210  RETURN
5220  DATA 104, 168, 104, 166, 223, 154, 72, 152, 72, 96
5230  PRINT D$"PR#1"
5260  PRINT """PR#1"RN
5270  RETURN
5280  PRINT D$"PR#1"
5300  RETURN"132N"
```

System of the Month

Ohio Scientific C3-C

By Tom Fox, Systems Editor

The C3-C is the latest and most capable computer yet from a company who announces new products at a pace that is sometimes hard to follow. Challenger I begat Challenger II, which soon gave way to the Challenger III series, of which the C3-C is one of at least four sub-models. Continuing the count, the C4 and C8 computers reverse the trend of increasing size with greater model numbers — they are desktop "educational" units. The C3-C, then, is the top of the line from this experienced microcomputer maker. All of Ohio Scientific's machines sport what is essentially the same Central Processing Unit (CPU), so across-the-board software compatibility is promised — and very nearly accomplished.

A PERFECT "3"

When our microcomputer industry was still a young one, Ohio Scientific carried the distinction of making the only "three-in-one" computer. They are still the only ones that do, which means either that Ohio Scientific remains smarter than their competition, or the idea wasn't a good enough one to copy. This unique idea was conceived during a time of indecision when the "best" microprocessor chip had yet to emerge (a situation that is hardly resolved even today). With a bold stroke, Ohio Scientific produced a computer with three separate 8-bit chips: the Z-80, the 6800 and the 6502A. The board housing these three devices and their supporting circuitry features a software-controllable "gear shifter" to activate any one of the microprocessors while disabling the two unused ones. It is important to recognize that the three chips can't work together in a cooperative mode, more's the pity.

Why such an extravagance? The manufacturer will tell you that it triples the amount of applications software that will run on the machine. In other words, it broadens the territory over which you can search for useful programs. The passage of time and the onset of harsh reality have exposed this concept to be somewhat illusionary, however. Business users find that they tend to stick with a package of interacting programs that are designed to work with each other, and, of course, to run on a single type of computer. It's highly unlikely, for example, that you will find an Accounts Receivable program running on the Z-80 that creates data files which are compatible with a General Ledger program written for the 6800 microprocessor.

It is not surprising that Ohio Scientific themselves have tended to focus their software efforts on one of the three chips: the 6502A, as it turns out. Their latest in-house developments are based on their own OS-65/U operating system, which runs only on the 6502A portion of the CPU board. The multi-user capability and hard disk drive, for example, are both incompatible with any programs yet available on the 6800 and Z-80. These two micros remain, however, as a bonus to the purchaser and reminder of Ohio Scientific's beginnings.

HARDWARE

The C3-C comes in a substantial rack enclosure that makes it look more like a $50,000 minicomputer than the microcomputer it really is. Other features, such as its 23-megabyte (23 million-character) hard disk drive, makes you wonder just what differences remain between minis and micros these days. The four-foot-high cabinet houses three drawer-slide chassis containing — from top to bottom — dual Shugart or Siemens floppy disk drives; a card cage containing CPU, memory and interface electronics; and one of the new Shugart SA-4008 Winchester-type disk drives.

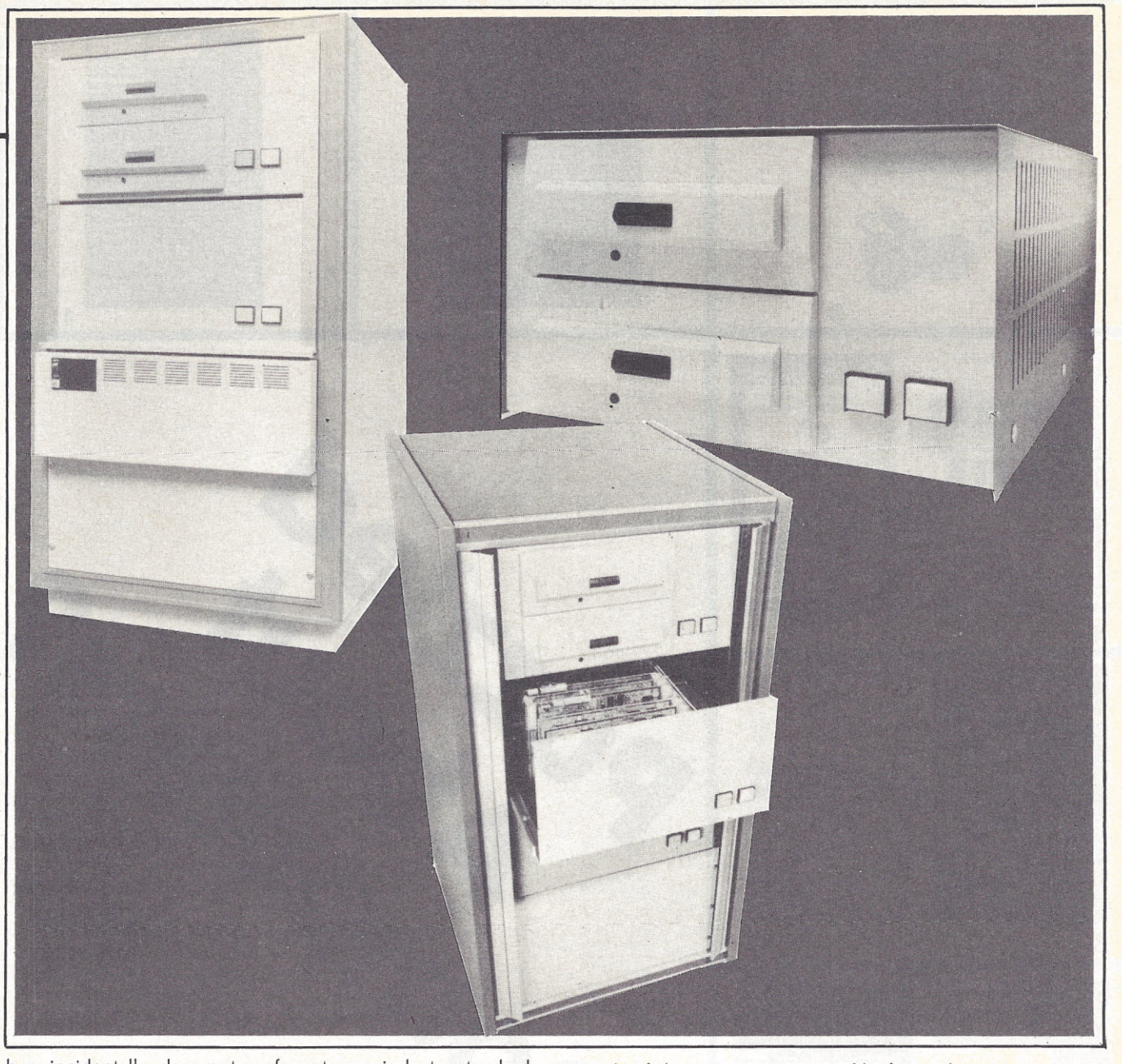
The SA-4008 is one of the new generation low cost and high performance hard disk drives that promises a new dimension for mass data storage in small business computers. The version utilized on the Ohio Scientific unit features 23 megabytes of usable space (of 29 megabytes total — all disks contain a certain amount of "housekeeping" information that can't be used for actual data storage). The rotating disk itself is enclosed within a dust-tight hermetic seal, reducing greatly the problems that can result in lost or contaminated data.

This means, though, that disks cannot be changed by the system operator. The most annoying result of this is that backup copies of your programs and data must be made onto floppy disks. It can take up to 100 8-inch floppies and several hours to completely back up a full disk. That's over $500 worth of diskettes.

A more subtle problem is that files larger than 230,000 bytes cannot be copied directly. For very large files, a program is needed to break up the data into smaller chunks and copy them, one at a time, onto floppy diskettes (and back onto the hard disk again, when needed). The storage and cataloging problem for such a stack of saucers could easily become a headache. Even though the use of double sided and/or double density floppies could reduce the problem by a factor of two or even four, the operational difficulties can still be very serious.

Lest you begin to wonder just how useful such an arrangement is, you can take heart in two thoughts: 1) Chances are fair that your application won't use more than 10 megabytes or so of the copious available capacity, and 2) At least one manufacturer has begun to make a more practical backup device. There will undoubtedly be more, since Ohio Scientific isn't alone in shouldering the tough Winchester backup problem. The one solution that has emerged comes from Massachusetts-based Alloy Engineering Company. It's a magnetic tape drive that can hold up to 12 megabytes in each of its 3M-type tape cartridges. Backing up the 23-megabyte disk can become a far more reasonable ten-minute, two-cartridge operation using this machine. The price is the only painful part: $3750 for a single-drive cartridge unit.

Looking into the main computer chassis, we see a 17-slot mother board that has plenty of spares for future expansion. The



bus, incidentally, does not conform to any industry-standard design — Ohio Scientific does not encourage non-factory additions to their computers. In addition to the three-processor CPU board, the drawer holds a floppy disk controller board, a two-card hard disk controller and a 48-kilobyte Random-Access Memory (RAM) board. The CPU board includes a single serial interface for the CRT terminal and an 8-bit parallel port that can be used for either input or output. Multi-terminal systems add a 16-port serial interface and a 48-kilobyte RAM card for each additional terminal. Also needed for multi-user operation is a separate four-kilobyte RAM board containing the multi-tasking executive program.

For special applications, you can install a 96-bit parallel input/output card or a dual digital-to-analog and analog-to-digital converter board. With a modest addition of parts, a real-time clock can be piggybacked onto the CPU board.

Your local dealer will add the printer and CRT terminal(s) needed to complete the package. Competition for these peripheral devices being what it is, it's easy to find some really excellent units at almost any computer store.

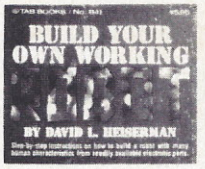
SOFTWARE

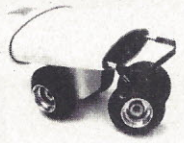
It all begins with the operating system, and Ohio Scientific complicates the picture by needing three of them: one for each of the microprocessors. No fewer than nine separate operating systems have found their way into the C3-C product line. Rather than attempt to relate a several-hundred-page catalog, we will simplify matters by concentrating on 6502A-based programs for this article, as the main thrust of Ohio Scientific's newest developments lies here.

The latest disk operating system is designated OS-65/U. Its companion is OS-65/D, which incorporates some additional features to aid in program development. As we were testing the C3-C, a new version of OS-65/U was released which allows the machine to run at 2 megahertz — twice its original speed. This should provide a dramatic performance improvement for existing programs. The disk operating systems operate in conjunction with the Level III multi-user executive.

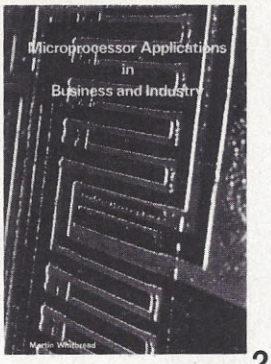
We were surprised at the manner in which memory space is allocated for these pieces of system software. A single four-kilobyte copy of the multi-user executive is always resident in a dedicated memory space, whereas each of the multiple users must squeeze in a separate 24-kilobyte copy of the OS-65/U disk operating system in its own 48-kilobyte "page" of RAM. The multi-user executive passes control of the microprocessor around between the users each 1/50th of a second, with a separate single-user operating system being dedicated for each terminal.

Puzzled About Computers?

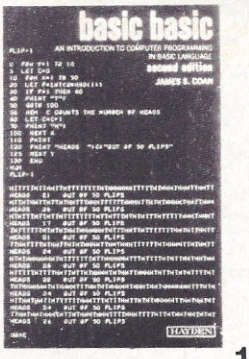




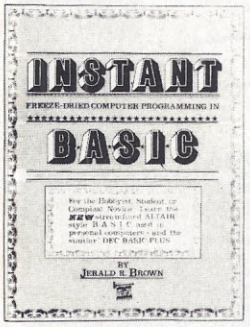
3



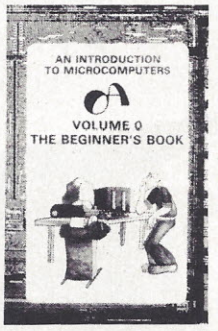
2



1



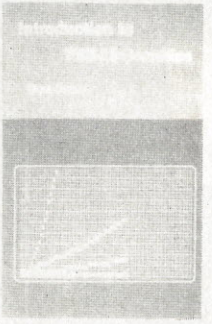
4



5



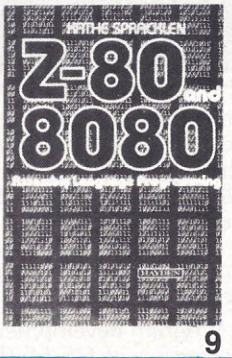
6



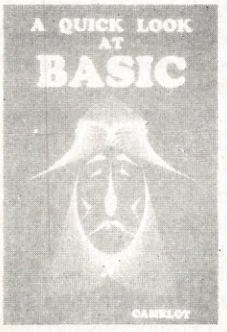
7



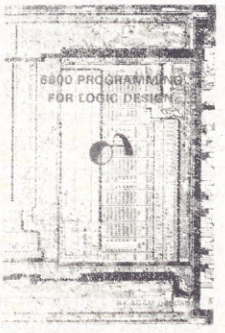
8



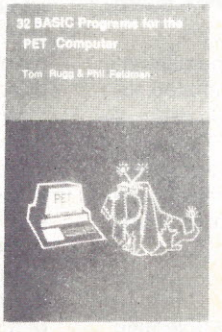
9



10



11



12



13

Data Dynamics Technology has a library of answers . . .

1 **BASIC: An Introduction to Computer Programming in BASIC Language -Second Edition**
By James S. Coan. 288 pages, $8.95
The author uses over 100 sample programs to illustrate the essential techniques of the language and to integrate BASIC programming with mathematics. Each language statement or capability is clearly explained at the time that it is first used in a sample program. Every section is followed by practice problems.
Order No. HAY5106-9, paper.

2 **Microprocessor Applications in Business and Industry**
By Martin Whitbread
160 pages, $18.00
This book provides an up-to-date introduction to micro-electronics, microprocessors and microcomputers. It explains the fundamentals of microprocessors and how they may be used to reduce administration and manufacturing costs, increase efficiency and assist in the development of new products.
Order No. CHP, paper.

3 **Build Your Own Working Robot**
By David L. Heiserman
234 pages, $5.95
Here are complete instructions, plans, schematics, logic circuits, and wiring diagrams for building Buster, the mechanical pet robot. He'll serve you coffee or bring you the morning papers. He'll forage for his own "food" and scream when he can't find it. His "curiosity" will get him into one plight after another, but Buster has the capacity to get himself out of trouble just as easily as he got into it!
Order No. TB841, paper.

4 **Instant BASIC**
By Jerald R. Brown
159 pages, $9.95
Written for the inexperienced, this activity oriented book will help you teach yourself microcomputer BASIC, and the similar DEC BASIC PLUS for programming your personal computer. There's never a dull page and plenty of activities. For those who already know some BASIC, this workbook can teach you the new microcomputer dialect.
Order No. DMX04-3, paper.

5 **An Introduction to Microcomputers: Volume 0 The Beginner's Book**
By Adam Osborne, 300 pages, $7.95
Here is the complete book for the beginner in the field of microcomputers. It describes component parts of a microcomputer system and relates them to the individual hobbyist.
Order No. OSB26-8, paper.

6 **6502 Applications Book**
By Rodnay Zaks
288 pages, $12.95
This book will teach you how to connect a board to the outside world and implement practical applications for the 6502. You will learn techniques ranging from simulated traffic control to analog-digital conversion.
Order No. C302, paper.

7 **Introduction to TRS-80 Graphics**
By Don Inman, 175 pages, $8.95
Here is a book that will explain the basics of graphic programming using real examples which can be run on the TRS-80. The book begins with some basic concepts like line drawing and leads on to more advanced concepts, such as moving figure animation.
Order No. DP18-5, paper.

8 **Programming the 6502**
By Rodnay Zaks
300 pages, $10.95
This book is designed to teach programming using the 6502. It will bring you to a point where you can write complete programs, as well as explain the advantages and disadvantages of the 6502.
Order No. C202, paper.

9 **Introduction to 8080 and Z-80 Assembly Language Programming**
By Kathe Spracklen, 192 pages, $7.95
Here is the first book that gives an introductory look at assembly language programming for the 8080 and Z-80. It will answer any questions an applications programmer may have on how to get the most out of his or her machine.
Order No. HAY 5167-0, paper

10 **A Quick Look At BASIC**
By Donald D. Spencer
64 pages, $3.95
A short, to-the-point introduction to the BASIC programming language. It is intended to serve as a self-instructional book designed for either the reader who has little or no knowledge of computers or computer programming, or the person who wants to learn BASIC quickly. This text introduces the student to the essentials of BASIC in a clear and simple manner. The emphasis is on giving the student the ability to use the computer to solve problems as quickly as possible.
Order No. CAM1008, paper.

11 **6800 Programming for Logic Design**
By Adam Osborne
310 pages, $9.50
This book explains how an assembly language program within a microcomputer system can replace combinatorial logic — that is, the combined use of "off-the-shelf," nonprogrammable logic devices.
Order No. OSB05-5, paper.

12 **32 Basic Programs for the PET Computer**
By Tom Rugg and Phil Feldman, 370 pages, $15.95
This book is for the computerist who would like to learn more about the PET 2001. It offers 32 chapters of fully documented, different programs such as games, math and other educational programs plus many more.
Order No. DP25-8, paper.

13 **INTERFACE AGE Binders and Slip Cases**
Data Dynamics Technology is now offering deluxe binders and slip cases which will place each back issue of INTERFACE AGE at your fingertips. Each binder is constructed of brown vinyl with INTERFACE AGE stamped in gold foil on the front cover and spine. The slip cases are identical, but in blue. These rugged binders and slip cases can hold 12 issues each and will protect your back issues of INTERFACE AGE for years.

4/80

DATA DYNAMICS TECHNOLOGY P.O. Box 1234, Cerritos, CA 90701

Name (Print)_____
Address_____
City _____ State_____ Zip_____

Please send me:

Description	Qty	Price	Total	Book Order #	Qty	Price	Total
Binders		7.50					
Slip Cases		5.95					

Shipping & Handling Charges
Binders and Slip Cases $1.50 ea. U.S., $2.00 ea. Foreign
Books $.75 ea. U.S., $1.50 ea. Foreign

TOTAL ORDER $_____
***TAX $**_____
SHIPPING & HANDLING $_____
TOTAL ENCLOSED $_____

[] master charge [] AMERICAN EXPRESS Cards [] VISA # _____ [] Check or M.O.(U.S. Funds drawn on U.S. bank)

Exp. Date_____ Signature _____

For European delivery contact **INTERFACE AGE Europe,** Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany.
*California residents add 6% sales tax. Availability and prices quoted subject to change without notice.
Please allow six weeks for delivery. You may photocopy this page if you wish to keep your INTERFACE AGE intact.
Orders *cannot* be shipped unless accompanied by payment, including shipping & handling and tax where applicable.
DATA DYNAMICS TECHNOLOGY, A Division of INTERFACE AGE Magazine (213) 926-9544

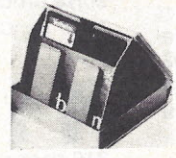
Good Buys From Disks, Etc.

Verbatim MINI-DISKS 10 for $29.00 or 3 for $10.00

Scotch or Memorex 10 for $31.00

Verbatim Digital Cassettes
For top performance in popular personal computers, including Apple, PET, Heathkit, Atari and Mattel.
2 for $5.45





Mini-Disk Protectors Fit 3-Ring Binders
Tough, glare-proof vinyl. Holds 2 diskettes and ID cards.
75¢ ea., 10 for $5.75

Storage Cases
Made of durable, molded plastic. Choice of black or beige.
Maxi Case, $4.25
Mini Case, $3.75

MasterCharge, Visa, Checks OK. No COD's.

Disks, Etc. P. O. Box 327 Center Valley, Pa. 18034

CIRCLE INQUIRY NO. 22

TRS-80 SOFTWARE 5110

★ C.P.A. General Ledger ★
★ Municipal Utility Billing (quote) ★
★ Municipal Budgetary Accounting ★
★ Balance-Forward Accounts Receivable ★
★ Open-Item Account Receivable ★
★ Contractor Job Costing ★
★ Accounts Payable ★
★ Restaurant Payroll ★
★ Farm Payroll — Union Payroll ★

INDEX SEQUENTIAL RANDOM ACCESS PROCESSING
TRS-80 $240.00 ea. IBM-5110 $520.00 ea.
ALTOS $410.00 ea.

For sample reports send $6.00
VISA, Mastercharge — OK —

COMPUTER SYSTEMS DESIGN, INC.

P.O. Box 735
Yakima, Wa. 98907
Call 1-509-575-0320

CIRCLE INQUIRY NO. 87

The arrangement is certainly less elegant than a true multitasking operating system, and terribly wasteful of memory space, but it seems to work. When we asked if different users could run under separate microprocessor/operating system combinations — OS-65/U on the 6502A at the same time as CP/M on the Z-80, for example — the factory replied, "theoretically."

Ohio Scientific also supplies an extended version of Microsoft's BASIC interpreter. The extensions involve a compatibility with the file structures in the new hard disk and a "money mode" of computation for increased accuracy in financial programs. The OS-65/D version of the operating system includes a text editor and line resequencing utility for BASIC programs.

With regard to applications programs, Ohio Scientific's catalog is bulging with a somewhat disorganized collection of games, word processors, database managers and general business programs that have been developed over the years for their earlier single-user systems. These programs run on one or another of the three on-board microprocessors, and — for the most part — have been collected from a variety of independent software houses. Only a few of the business programs have been adapted to take advantage of the multiuser capabilities of the C3-C, but we expect that the selection will improve with the passage of time.

For 6502A machine-language programmers, Ohio Scientific includes an assembler and various programming utilities. A demo diskette containing games and various programming examples is thrown in for good measure. Finally, under the software heading, an enhancement in the Level III executive makes it possible to interconnect a number of C3s together via a communications network. This exposes a really exciting capability, even though the applications programming for such an assemblage would definitely be an ambitious project. If you need distributed processing, though, Ohio Scientific is one of the few places you can find it at these prices.

A CLOUD OVER OHIO

The microcomputing industry is still too small to keep a secret for long, and we've heard around and about that this midwest supplier's reputation for after-sales support has been less than exemplary. In tracking it down, we found one otherwise very successful dealer who abandoned this manufacturer after months of frustration with Ohio Scientific's customer service and repair department. The documentation, this dealer claimed, was so poor that daily telephone calls to the factory were necessary to keep their customers' systems operating; and it was difficult to entice a knowledgeable factory representative to answer the phone. At the other extreme, we spent some time with another dealer, not 20 miles away, who ships many Ohio Scientific systems every month, and who couldn't praise the manufacturer enough. The documentation, he said, is undergoing a transformation for the better. "Poor getting tolerable," he phrased it. He is eagerly looking forward to his first copy of a Sams PhotoFact manual on the C3-C hardware.

What could be the reasons for two such disparate judgments? One is timing. We have seen evidence that Ohio Scientific is beginning to catch up in those areas where they might have been delinquent in the past: customer support, documentation and training. Another answer might be a simple human one: Perhaps the chemistry has to be "right" between a manufacturer and its retail outlet.

Our advice is to shop around carefully before you invest $10,000+ into a computer like the C3-C. But that's exactly the same counsel we would make regarding any other manufacturer's product, even one from an industry giant. If the store you're dealing with doesn't think their computer is the best one in the world for your needs, find another computer — or find another store.□

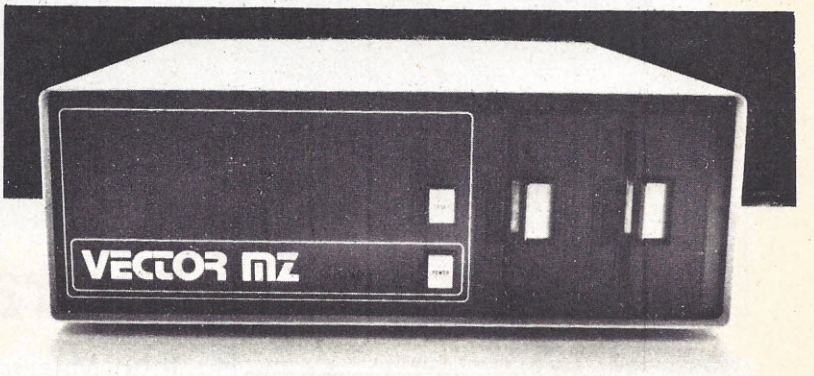
OVER $10,000 Value!

A Complete Business System

only $5995

Price includes air freight shipping





VECTOR SYSTEM B, complete with Vector Mindless Terminal, 64K of RAM, Dual Floppy Disks (630 kilobytes of storage), and printer . . .
so complete, you'll get all cables, box of 10 floppy disks, and *EVEN* a box of 3500 sheets of fanfold paper.

OVER $3500 OF SOFTWARE INCLUDED!!! Digital Research's 2.0 CP/M Disk Operating System, Microsoft 80 BASIC Interpreter

AND one of the finest Business Packages — from Retail Science's PEACHTREE SOFTWARE, which includes:



- General Ledger
- Accounts Receivable
- Accounts Payable
- Inventory
- Payroll

System may be expanded for multi-user time-sharing data and word processing — up to 5 terminals — at nominal cost!

The System B doubles as an *excellent* Word Processing System.
Software at slight additional cost.

Third-party maintenance agreements being negotiated.

CIRCLE INQUIRY NO. 94

MiniMicroMart, Inc.

1618 James Street, Syracuse, NY 13203 (315) 422-4467 TWX 710-541-0431

Powered Down Bipolar PROMs

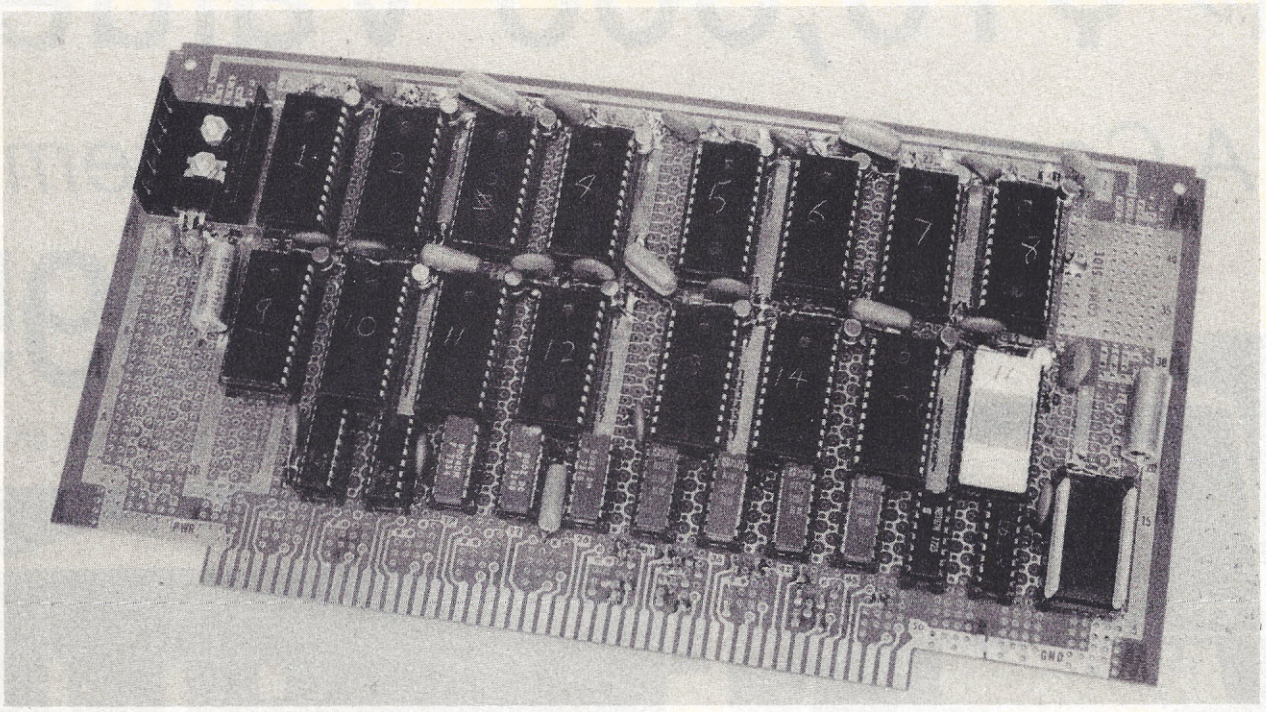


PHOTO 1

A "Cool" Operating System

By John MacDougall

Many beginning computerists are dismayed to find that when they get their system operational, they spend much of their time loading and unloading cassettes or paper tape. This program offers a chance to be able to call up the text editor or BASIC at the touch of a key, or to turn on the machine and immediately start to write.

The method is surprisingly easy. The secret is powered down bipolar PROMs.

FUSIBLE LINK PROMS

Fusible link PROMs are made by a number of manufacturers. Their main distinguishing features are the speed with which data can be addressed and the method of forming the fusible links. Since the speed of these devices is usually far faster than most microprocessors, this parameter is of little interest.

The fusible links are formed out of a metal such as nichrome or from a doped polysilicon material. Either way, the basic technique is to provide a controlled amount of excessive current to the bit of choice, thus blowing the link and permanently forming the bit into a zero (or one, depending upon the manufacturer).

This article will be concerned with the nichrome isoplanar PROM with tristate outputs (93448) made by Fairchild. This is a 4K bit PROM arranged in 512 x 8 configuration. The 93448 is one of the easiest to program, especially with respect to the hobbyist. All that is needed is that the appropriate address be selected, the chip disabled, and then 20 volts applied to the output bit to be programmed.

Furthermore, the 93448 distinguishes itself from similar MOS EPROMs by the speed with which it can be brought up to operating voltage and the data read. This high speed (usually ten times as fast as a similar EPROM) and the single five volt power supply required for operation is the secret to operation in the power down mode. These devices are so fast that power down operation is possible even in ten megahertz systems.

Of course, there is a disadvantage. The software must be perfect before a fusible link PROM is burned in since once burned, these devices are permanently programmed and there is no way that anyone, even the original manufacturer, can recover the device.

POWER-DOWN OPERATION

Operation in the power-down mode means just that — fully powered-down. There is no partial or standby power; the de-

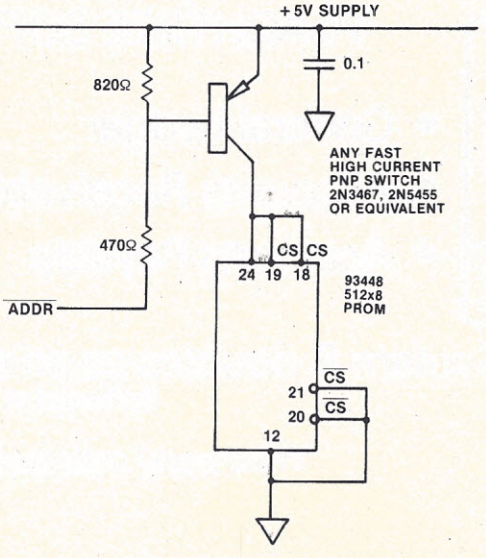


Figure 1. Powered-Down PROM Circuit Configuration

vices are really turned off. To do this it is necessary to arrange an active switch in the power lead of the PROM and turn the power back on when the PROM is needed. This is where the single five volt supply and the high speed of the bipolar PROM are important. The circuit used for power down operation is shown in Figure 1.

A PNP high speed switch is placed in the power supply lead and also connected to the high enable inputs. The low enable inputs are grounded. They could also be used in some form of separate chip select. In the diagram the only mode of chip select is by means of the power switch so that when the PROM is addressed, it is powered up, activated, and ready to go. When the PROM is not selected, it is completely powered down.

POWER-DOWN PROMS — A TYPICAL SYSTEM

The system used here is Z-80 based and uses the TDL "ZAPPLE" Monitor. This monitor resides at F000 and above. The system diagram for the monitor board is shown in Figure 2. The inputs are first buffered and partially decoded, after which they are used to drive a 74LS42 (BCD to decimal) decoder. The buffering is needed for other circuitry which is also on the same board as the monitor system.

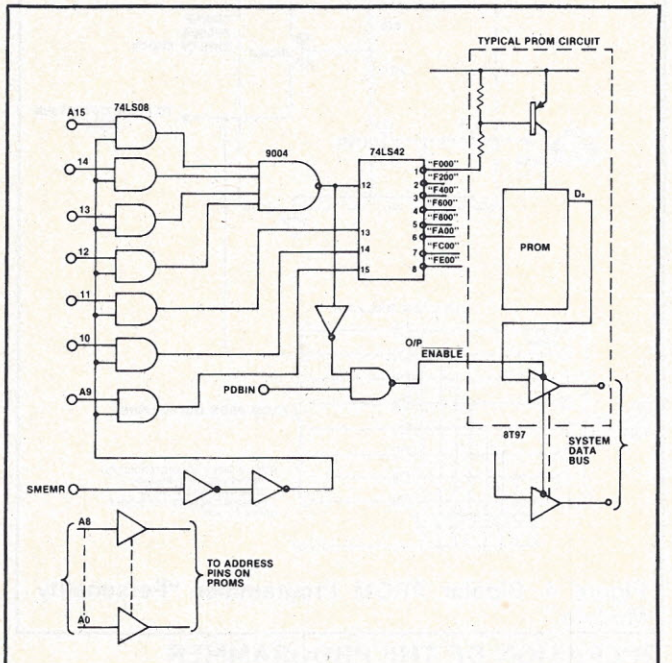


Figure 2. Schematic Circuit Diagram for a Powered-Down PROM Monitor Board.

The outputs of the 74LS42 are low active. This is just perfect for driving the PNPs, which are used to power up the PROMs as shown. Since these PROMs are bipolar, they have excellent output capability and will drive the data bus directly in a small system. However, this system is getting bigger and bigger, and separate board bus drivers are becoming very important.

In addition to the Monitor, my system has 8K BASIC and a text Editor, all operating out of powered down PROMs. It's beautiful and the regulators don't even get warm.

POWER

The best place to consider power is with a BASIC board. An 8K BASIC is used here. No matter how this is implemented in a system, it would take five volts at about 140 mA for buffering and decoding. If I must have extra RAM for this and load from tape, the memory would require about 1.9 amperes if 21L02s were used, or perhaps 700 mA if transparent refresh 4K dynamics were used. In either case, the regulator(s) would get hot.

With powered down PROMs, the situation is vastly different because they draw no power at all until accessed, and then only one PROM is on at a time. In the very worst case, when only one PROM is on for about ¼ of the time, the current drawn by the memory part of my board would add an additional 40 mA to the total, giving a final power consumption of 180 mA times five volts. The regulator doesn't even get warm and most of that is caused by the buffers and decoders.

One note of caution after all this. Each PROM draws about 150 mA during operation. This is a huge transient to throw on the regulator and system power supply. For the system to work at all, put in lots of bypass capacitance. I use 0.1 uF across the five volt supply at each PROM.

Photo 1 shows an 8K BASIC board with the 16 PROMs, the PNP power-up drivers and the single 5 volt regulator.

PROGRAMMING THE PROMS

All of the information for programming Fairchild Bipolar PROMs is contained in the appendix following this article. Briefly, bipolar PROMs of the 93448 type are programmed by pulsing each output in turn with a current-limited 20.5 volt pulse once for each bit which is to be a zero. Note the single iteration as compared to EPROM programming and also the bit by bit programming. As with all PROM programming, the specifications are critical and should be followed carefully. A summary of the programming specifications for the 93448 is contained in Table 1.

Table 1. Programming Specifications

Parameter	Recom. Value	Units	Comments
Address input	VIH 5.0 VIL 5	V V	Do not leave inputs open
Chip Select	5.0	V	Pin 13 or 14 or Both
Programming Voltage Pulse	20.5	V	Applied to output to be programmed
Programming Pulse Width	180 μs		All bits can be programmed in ⩽1.0 seconds
Duty Cycle, Programming Pulse	20	%	Maximum duty cycle to maintain TC <85 °C
Programming Pulse Rise Time	1.0	μs	
Power Supply Voltage	5.0	V	
Case Temperature	25	°C	
Programming Pulse Current	Max. 100	mA	If pulse generator is up set current limit to the max value
Number of Pulses	8		
LVCC Read	4.4	V	Programming read verify

Commercial PROM programmers are very expensive, and yet most of the hardware in the programmer is already present in most microcomputer systems. The real trick for maximum versatility is to have a universal type of interface, build a small personality module and keep the control timing and other factors in the system software. The circuit diagram for the programmer here is given in Figures 3 and 4. This was wired up on a Solid-State Music I/O PROM-Universal card which had the decoding and some of the 8212 circuitry already in place.

The parts shown in Figure 4 are specific to the bipolar PROM. With a different setup they could be easily modified so that EPROMs could be programmed as well as Bipolar PROMs. Note again that the methods of programming these two types of PROMs are completely different. Although their

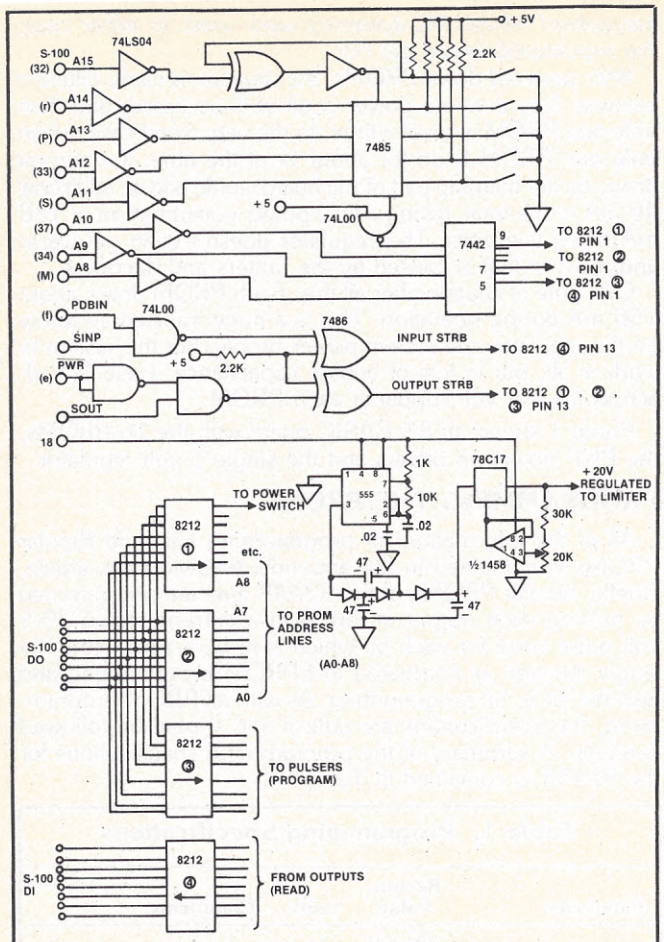


Figure 3. PROM Programmer Schematic, exclusive of "Personality Module."

address and data pinouts are identical, they cannot be programmed in the same setup or with the same software drivers.

The programmer also has an on-board oscillator and voltage doubler to provide the necessary high voltage for the programming pulses. This doubler with appropriate change of voltage could also be used for programming EPROMs, for instance.

The reason for having the address and data set up through I/O ports was to give maximum flexibility to the use of the programmer and to allow as much of the programming to be performed in software as possible.

A complete listing of the software necessary for driving the programmer is presented at the end of this article. The listing is in Technical Design Labs relocatable format. Thus, all addresses marked ' are relocatable and must be modified if the program resides at other than 00. Addresses not marked with an apostrophe are absolute and should not be modified.

The listing contains calls to a number of routines which reside in the monitor. The monitor is a modified TDL "ZAPPLE" monitor, but the locations are correct for the original version. The monitor is located at F000H as suggested by TDL and is in powered-down PROM, of course. If one wishes to fit this program to a different monitor, a more complete description of these calls is presented below:

F000	Monitor start location
F440	Monitor routine for printing a message whose location is given in register pair HL and length in register B
F470	Routine to print register pair HL (in HEX) plus a space
F504	Routine to print a CR plus a LF
F50C	Routine to accept any keyboard input
F533	Routine to get 1 parameter (address) from the keyboard and place it in the top of the stack
F582	Routine to print one byte in HEX
F730	Routine to get an ASCII character input from the keyboard

The program is arranged so that a number of subprograms can be addressed if the full programming schedule is not desired. These sub-programs are as follows:

Location Description

007D'	Check the PROM for all 1's (unprogrammed)
00DD'	Program a single byte in the PROM from the keyboard
00F5'	Dumps the PROM contents to RAM at location 1000H

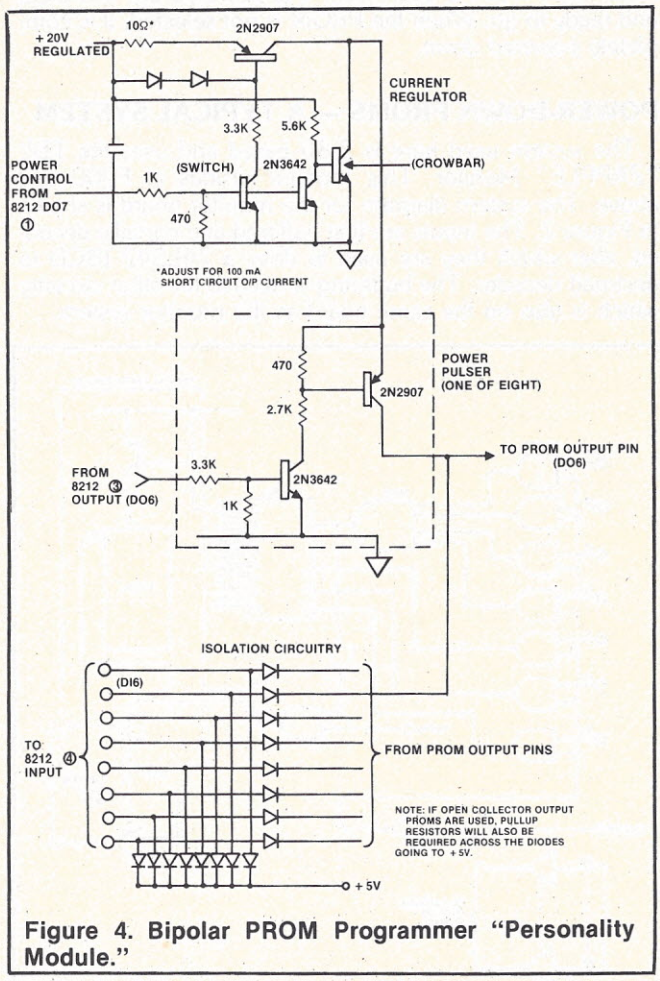


Figure 4. Bipolar PROM Programmer "Personality Module."

OPERATION OF THE PROGRAMMER

First the programmer is connected to the computer. In the system used here, this is done by plugging it into the S-100 bus extender card. Then the program is loaded at the desired location using the appropriate monitor load routine, and finally, the program is entered as a jump. The program will then type the message "insert PROM." The PROM is then put into the socket and CR hit.

The program checks the PROM to see if there are any programmed locations. The checking can be terminated by hitting control-C at any time. If the PROM is good, the system waits a decision to continue (CR) or abort (Cont-C). A carriage return signifies the intention to continue and program the PROM, so the next message is "start addr?". This is asking for the start of the RAM location whose contents are to be burned into the PROM. When this is entered, followed by a CR, the PROM is programmed, and the system returns to the monitor.

As a final safeguard, the program also checks the PROM contents, as programmed, against the RAM. However, if you wish, you may also transfer the PROM contents to RAM at location 1000H by entering the program at location 00F5'. This is also useful for reading the contents of an unknown PROM.□

PROGRAM LISTING

```
                        .TITLE 'ASSEMBLY LISTING FOR Z-80 PROM PROGRAMMER'
                        .RADIX 16
                        ;This program is written in TDL Z-80 relocatable
                        ;code and is intended for use  with the fusible
                        ;link PROM programmer.  John MacDougall Nov 1977
                        ;
                        ;MAIN PROGRAM follows
0000' 3E00                      MVI     A,00    ;SET PROGRAMMING PORT = 0
0002' D30B                      OUT     0B      ;AND REMOVE POWER
0004' D308                      OUT     08      ;FROM PROGRAM SUPPLY
0006' 21 00C4'                  LXI     H,MSG1  ;MESSAGE LOCATION
0009' 060D                      MVI     B,0D    ;MESSAGE LENGTH
000B' CD F440                   CALL    0F440   ;MON. ROUT. TO PRINT MSG
000E' CD F730   INPUT:  CALL    0F730   ;MON. ROUTINE FOR KBD INPUT
0011' FE0D                      CPI     0D      ;KEEP LOOKING FOR CR
0013' 20F9                      JRNZ    INPUT   ;
0015' CD F504                   CALL    0F504   ;MON. ROUTINE FOR CR+LF
0018' CD 0059'                  CALL    FFCHK   ;LOOK FOR ALL 1'S IN PROM
001B' CD F730                   CALL    0F730   ;KBD INPUT
001E' FE03                      CPI     03      ;CONTROL C ABORTS
0020' CA F000                   JZ      0F000   ;TO THE MONITOR RESTART
0023' CD F504                   CALL    0F504   ;OTHERWISE DO A CRLF
0026' 21 00D1'                  LXI     H,MSG2  ;MESSAGE LOCATION
0029' 060C                      MVI     B,0C    ;LENGTH
002B' CD F440                   CALL    0F440   ;PRINT IT
002E' CD F533                   CALL    0F533   ;MON. ROUTINE FOR 1 PARAM.
0031' E1                        POP     H       ;STARTING ADDRESS IN HL
0032' 22 0057'                  SHLD    ADDR    ;STORE STARTING ADDRESS
0035' 11 0000                   LXI     D,0     ;PROM ADDRESS STARTS AT 00
0038' 4E        PRO:    MOV     C,M     ;GET A BYTE FROM MEMORY
0039' CD 0083'                  CALL    PROG    ;CALL PROG. TO BURN A BYTE
003C' 13                        INX     D       ;NEXT PROM ADDRESS
003D' 23                        INX     H       ;NEXT MEMORY ADDRESS
003E' 7A                        MOV     A,D     ;READY FOR ENDCHECK
003F' FE02                      CPI     02      ;200H ENDS 512 X 8 PROMS
0041' 20F5                      JRNZ    PRO     ;KEEP PROGRAMMING TIL DONE
0043' 2A 0057'                  LHLD    ADDR    ;GET THS MEMORY START ADDR.
0046' 11 0000                   LXI     D,00    ;PROM STARTS AT ZERO
0049' 4E        CHK:    MOV     C,M     ;GET THE MEMORY BYTE
004A' CD 00B1'                  CALL    CHECK   ;CHECK AGAINST THE PROM
004D' 13                        INX     D       ;NEXT LOCATION
004E' 23                        INX     H       ;IN PROM AND MEMORY
004F' 7A                        MOV     A,D     ;LOOK FOR END
0050' FE02                      CPI     02      ;DONE WITH 4K PROM?
0052' 20F5                      JRNZ    CHK     ;NO--KEEP CHECKING
0054' C3 F000                   JMP     0F000   ;WHEN DONE GO TO MONITOR
                        ;The following is space for memry start address
0057'                   ADDR:                   ;STARTING ADDRESS
0057'                   .BLKB 2
                        ;
                        ;This subroutine checks for  1's in all locations
0059' 21 0000   FFCHK:  LXI     H,00    ;PROM ADDRESS STARTS AT 0
005C' CD F50C   KBD:    CALL    0F50C   ;ANY KBD  INPUT
005F' B7                ORA     A       ;STOPS THE CHECKING
0060' C0                RNZ             ;AND RETURNS
0061' 7C                MOV     A,H     ;ADDRESS TOP BYTE
0062' D306              OUT     08      ;TO THE PROM
0064' 7D                MOV     A,L     ;ADDRESS BOTTOM BYTE
0065' D309              OUT     09      ;TO THE PROM
0067' DB0B              IN      0B      ;GET THE PROM BYTE
0069' FEFF              CPI     0FF     ;ALL ONE'S?
006B' 2809              JRZ     CONT1   ;YES, SKIP THE ERROR MSG
006D' F5                PUSH    PSW     ;STORE THE BYTE TEMMPORILY
006E' CD F470           CALL    0F470   ;MON. ROUT. PRINTS ADDR+SP
0071' F1                POP     PSW     ;GET THE BYTE BACK
0072' 4F                MOV     C,A     ;READY TO PRINT IT
0073' CD F582           CALL    0F582   ;PRINT THE BYTE
0076' 23        CONT1:  INX     H       ;NEXT ADDR
0077' 7C                MOV     A,H     ;TO CHECK FOR END
0078' FE02              CPI     02      ;DONE WITH 4K PROM?
007A' C8                RZ              ;YES, BACK TO THE MAIN PROG
007B' 18DF              JMPR    KBD     ;KEEP GOING
                        ;This is a small entry point routine to check
                        ;PROM for one's in all locations
007D' CD 0059'  CHECK1: CALL    FFCHK   ; FOR ONE'S IN ALL LOCNS
0080' C3 F000           JMP     0F000   ;RETURN TO THE MONITOR
                        ;
                        ;This is the main programming sub-routine
0083' E5        PROG:   PUSH    H       ;SAVE THE
0084' D5                PUSH    D       ;ADDRESS REGISTERS
0085' 7A                MOV     A,D     ;SET UP
0086' D308              OUT     08      ;THE TOP PROM ADDRESS
0088' 7B                MOV     A,E     ;AND
0089' D309              OUT     09      ;THE BOTTOM ADDRESS
008B' 0601              MVI     B,01    ;SET THE SHIFT BIT
008D' 7A                MOV     A,D     ;TOP ADDRESS
008E' F680              ORI     80      ;SET BIT TO TURN ON POWER
0090' D308              OUT     08      ;AND  DO IT
```

```
0092' 79          BYTE:   MOV     A,C     ;GET THE BYTE
0093' 2F                  CMA             ;READY TO PROGRAM THE 0'S
0094' A0                  ANA     B       ;SELECT ONE ZERO ONLY
0095' D30B                OUT     0B      ;BURN IT IN
0097' CD 00A8'            CALL    TIME    ;FOR  THE ALOTTED TIME
009A' 3E00                MVI     A,00    ;TURN OFF THE
009C' D30B                OUT     0B      ;DATA PORT
009E' CB20                SLAR    B       ;SHIFT LEFT FOR NEXT PROG
00A0' 30F0                JRNC    BYTE    ;DO FOR THE NEXT ZERO
00A2' 7A                  MOV     A,D     ;DONE, NOW
00A3' D308                OUT     08      ;      TURN OFF THE POWER
00A5' D1                  POP     D       ;RECOVER THE
00A6' E1                  POP     H       ;           ADDRESSES
00A7' C9                  RET             ;BACK TO RAM FOR NEXT BYTE
                  ;This is the pulse timing loop.
                  ; It is set here for 8 loops
                  ;of 22.5 us each thus giving a  pulse of >180 us.
00A8' C5          TIME:   PUSH    B       ;SAVE THE SHIFT POSN
00A9' 0608                MVI     B,08    ;8 LOOPS
00AB' E3          EXCH:   XTHL            ;EXCHANGE--LOTS OF TIME
00AC' E3                  XTHL            ;AGAIN -- CEPT TAKES 22.5US
00AD' 10FC                DJNZ    EXCH    ;KEEP GOING FOR 8 TIMES
00AF' C1                  POP     B       ;RECOVER THE SHIFT  POSN
00B0' C9                  RET             ;BACK FOR ANOTHER ZERO
                  ;
                  ;This subroutine compares a PROM byte with memory
                  ;and prints the address and error byte .
00B1' 7A          CHECK:  MOV     A,D     ;ADDRESS
00B2' D308                OUT     08      ;THE PROM
00B4' 7B                  MOV     A,E     ;
00B5' D309                OUT     09      ;
00B7' DB0B                IN      0B      ;GET THE PROM BYTE
00B9' B9                  CMP     C       ;COMPARE  THE MEMORY BYTE
00BA' C8                  RZ              ;RETURN IF EQUAL
00BB' F5                  PUSH    PSW     ;SAVE THE PROM BYTE
00BC' CD F470             CALL    0F470   ;MON. ROUT. TO PRINT ADDR.
00BF' F1                  POP     PSW     ;RECOVER BYTE
00C0' CD F582             CALL    0F582   ;PRINT THE PROM BYTE
00C3' C9                  RET             ;BACK FOR MORE
00C4'             MSG1:                   ;MESSAGE TO INSERT PROM
00C4' 494E534552  .ASCII "INSER"
00C9' 542050524F  .ASCII "T PRO"
00CE' 4D0D0A      .ASCII "M" [0D] [0A]
00D1'             MSG2:                   ;MESS. TO TYPE START ADDR.
00D1' 5354415254  .ASCII "START"
00D6' 2041444452  .ASCII " ADDR"
00DB' 3F20        .ASCII "? "
                  ;
                  ;This small program can be accessed to program
                  ; a single byte.
00DD' 3E00        SINGLE: MVI     A,00    ;INITIALIZE
00DF' D30B                OUT     0B      ;           THE PROG PORT
00E1' CD F533             CALL    0F533   ;MON. ROUT. FOR PROM ADDR
00E4' E1                  POP     H       ;TO HL FOR ECHO
00E5' CD F470             CALL    0F470   ;ECHO IT
00E8' EB                  XCHG            ;EXCHANGE HL & DE
00E9' E5                  PUSH    H       ;SAVE THE ADDRESS
00EA' CD F533             CALL    0F533   ;MON. ROUTINE FOR KBD BYTE
00ED' C1                  POP     B       ;RECOVER THE BYTE
00EE' E1                  POP     H       ;RECOVER THE ADDRESS
00EF' CD 0083'            CALL    PROG    ;BURN THE BYTE
00F2' C3 F000             JMP     0F000   ;BACK TO THE MONITOR
                  ;
                  ;This program can be accessed to dump the PROM
                  ;contents to RAM at location 1000H.
00F5' 11 1000     DUMP:   LXI     D,1000  ;RAM STARTING ADDRESS
00F8' 21 0000             LXI     H,00    ;PROM STARTS AT 00
00FB' 7C          ADDR1:  MOV     A,H     ;SET UP PROM
00FC' D308                OUT     08      ;           ADDRESS (HIGH)
00FE' 7D                  MOV     A,L     ;AND
00FF' D309                OUT     09      ;      (LOW)
0101' DB0B                IN      0B      ;GET THE BYTE FROM PROM
0103' 12                  STAX    D       ;STORE IN RAM
0104' 13                  INX     D       ;SET THE
0105' 23                  INX     H       ;           NEW LOCATIONS
0106' 7C                  MOV     A,H     ;ARE
0107' FE02                CPI     02      ;     WE DONE?
0109' CA F000             JZ      0F000   ;YES-- BACK TO MONITOR
010C' 18ED                JMPR    ADDR1   ;NO-- KEEP GETTING
                  .END

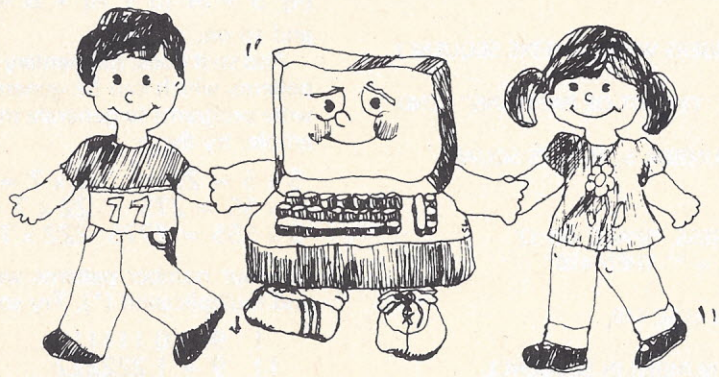

TDL Z80 RELOCATING ASSEMBLER VERSION 1.1
ASSEMBLY LISTING FOR Z-80 PROM PROGRAMMER
+++++ SYMBOL TABLE +++++


ADDR    0057'     ADDR1   00FB'     BYTE    0092'     CHECK   00B1'
CHECK1  007D'     CHK     0049'     CONT1   0076'     DUMP    00F5'
EXCH    00AB'     FFCHK   0059'     INPUT   000E'     KBD     005C'
MSG1    00C4'     MSG2    00D1'     PRO     0038'     PROG    0083'
SINGLE  00DD'     TIME    00A8'
```

My TRS-80 Likes Me

When I Teach Kids How to Use It

Part 8



By Bob Albrecht

Copyright 1979 by Bob Albrecht

RECAP

Last time we left you with a program for *you* to write. If you did so, perhaps some kids you know have been exploring number patterns such as the following.

(1) $11^2 = 121$
$111^2 = 12321$
$1111^2 = 1234321$
and so on

(2) $34^2 = 1156$
$334^2 = 111556$
$3334^2 = 11115556$
and so on

(3) $23^2 = 529$
$233^2 = 54289$
$2333^2 = 5442889$
and so on

(4) $98^2 = 9604$
$998^2 = 996004$
$9998^2 = 9996004$
and so on

(5) $12^2 = 144$
$102^2 = 10404$
$1002^2 = 1004004$
and so on

(6) $301^2 = 90601$
$302^2 = 91204$
$303^2 = 91809$
and so on

Each pattern is obtained by squaring each number of a sequence of numbers.

SEQUENCE (1): 11, 111, 1111, and so on.
SEQUENCE (2): 34,334,3334, and so on.
SEQUENCE (3): 23, 233, 2333, and so on.
SEQUENCE (4): 98, 998, 9998, and so on.
SEQUENCE (5): 12, 102, 1002, and so on.
SEQUENCE (6): 301, 302, 303, and so on.

For any sequence, let's use s_1 to stand for the first number, s_2 for the second number, s_3 for the third number, and so on. Hang on — here we go.

s_1 is always given.
$s_2 = m(s_1 + a) + b$
$s_3 = m(s_2 + a) + b$
and so on

Wait a minute! Where did m, a and b come from? OK, here are some examples.

SEQUENCE (1): $s_1 = 11$ $m = 10$ $a = 0$ $b = 1$
$s_1 = 11$
$s_2 = m(s_1 + a) + b = 10(11 + 0) + 1 = 111$
$s_3 = m(s_2 + a) + b = 10(111 + 0) + 1 = 1111$
It works. Let's try another one.

SEQUENCE (2): $s_1 = 34$ $m = 10$ $a = -1$ $b = 4$
$s_1 = 34$
$s_2 = m(s_1 + a) + b = 10(34 + (-1)) + 4 = 334$
$s_3 = m(s_2 + a) + b = 10(334 + (-1)) + 4 = 3334$
That works, too. Let's try one more.

SEQUENCE (6): $s_1 = 301$ $m = 1$ $a = 1$ $b = 0$
$s_1 = 301$
$s_2 = m(s_1 + a) + b = 1(301 + 1) + 0 = 302$
$s_3 = m(s_2 + a) + b = 1(302 + 1) + 0 = 303$
We leave SEQUENCES (3), (4) and (5) for you to do.

In case you haven't gotten around to writing the program we suggested last time, try ours. The following program:

1. READs the values of S, M, A and B, which define a sequence.

2. Shows the first term of the sequence *and* the square of the first term.
3. If the user presses the space bar, shows the next term and its square.
4. If the user presses 'Q' goes to (1) and starts a new sequence. Unless, of course, a flag (1E37) is encountered as the value of S.

Our program is much like Number Patterns #5, which we showed you last time.

```
100 REM***NUMBER PATTERNS #6
110 DEFDBL S, M, A, B
120 CLS

200 REM***READ FOUR NUMBERS WHICH DEFINE SEQUENCE
210 READ S, M, A, B
220 IF S = 1E37 THEN PRINT "I'M OUT OF PATTERNS" : END

300 REM***SHOW LATEST NUMBER, S, AND ITS SQUARE
310 PRINT S "*" S "=" S*S

400 REM***WAIT FOR KEY PRESS, 'SPACE' OR 'Q'
410 KEY$ = INKEY$: IF KEY$ = "" THEN 410
420 IF KEY$ = " " THEN 510
430 IF KEY$ = "Q" THEN 120 ELSE 410

500 REM***COMPUTE NEXT NUMBER IN SEQUENCE
510 S = M*(S + A) + B
520 GOTO 310

900 REM***VALUES OF S, M, A, B
910 DATA 11, 10, 0, 1
920 DATA 34, 10, -1, 4
930 DATA 23, 10, -1, 3
940 DATA 98, 10, 1, 8
950 DATA 12, 10, -2, 2
960 DATA 301, 1, 1, 0
970 DATA 1E37, 1E37, 1E37, 1E37     End of data flags

999 END
```

The preceding program, with appropriate choices of values of S, M, A and B, will let you explore:

- Arithmetic sequences
- Geometric sequences
- Mixed arithmetic — geometric sequences

Sometimes, there is more than one way to define a sequence in terms of S, M, A and B. For example, here are some *arithmetic* sequences.

SEQUENCE	S	M	A	B
1, 2, 3, ...	1	1	1	0
1, 2, 3, ...	1	1	0	1
2, 4, 6, ...	2	1	2	0
2, 4, 6, ...	2	1	0	2
2, 5, 8, ...	2	1	3	0
2, 5, 8, ...	2	1	0	3

Well, M seems to be always one (1). But, you decide whether to use A or B to get the next term in the sequence. Can M be other than one (1) to get an *arithmetic* sequence?

Let's try some "pure" geometric sequences.

SEQUENCE	S	M	A	B
1, 2, 4, 8, ...	1	2	0	0
2, 4, 8, 16, ...	2	2	0	0
10, 100, 1000, ...	10	10	0	0

A geometric sequence is determined by S and M, with A and B both zero. Now let's try some "mixed" arithmetic/geometric sequences.

```
* S = 1    M = 2    A = 3    B = 0
(1) S = 1
(2) S = M*(S + A) + B = 2*(1 + 3) + 0 = 8
(3) S = M*(S + A) + B = 2*(8 + 3) + 0 = 22
```

and so on.

Or, using the full power of S, M, A and B, we might get the following sequence.

```
* S = 1    M = 2    A = 3    B = 4
(1) S = 1
(2) S = M*(S + A) + B = 2*(1 + 3) + 4 = 12
(3) S = M*(S + A) + B = 2*(12 + 3) + 4 = 34
(4) S = M*(S + A) + B = 2*(34 + 3) + 4 = 78
```

and so on.

And so it goes. Elementary school math books show many patterns which can be generated by our program. *You* can write programs to generate many additional patterns. For example, try these:

```
5 x 5 = 25        2 x 7 = 14        9 x 5 = 45
5 x 55 = 275      22 x 7 = 154      99 x 5 = 495
5 x 555 = 2775    222 x 7 = 1554    999 x 5 = 4995
```

In our number patterns we have used only addition (+) and multiplication (*). Try some patterns using division (/).

```
  1  9 = 0.111111 ...
 11  9 = 1.222222 ...
111  9 = 12.3333  ...
1111  9 = 123.444  ...
```

and so on.

Remember! We have worked only with patterns in which the next number is a simple function of the previous number, using constants such as m, a and b.

NEXT TIME

This series began in the August/September, 1979, with verbal stuff and simple graphics. Next time, we will return to verbal and graphics patterns. Watch for fireflies, constellations, animated tiny creatures, games and other recreations.

We will show you excerpts from books-in-progress which explore recreational and educational uses of computers. Previews of things to come will appear in three magazines: *Interface Age*, *The Computing Teacher* and *Recreational Computing*.

The Computing Teacher is the professional journal of computing for teachers, published by the International Council for Computing Education (ICCE). *The Computing Teacher* is designed to help the classroom teacher in the classroom, resource center or other learning environment.

The Computing Teacher, c/o Dr. David Moursund
University of Oregon, Eugene, OR 97403

Recreational Computing is the magazine of recreational and educational computing. *RC* focuses on the pedagogy of games, the classroom of the future, the home as a learning center and, especially, computers as *Worlds of If* for people to explore.

Recreational Computing
P.O. Box E, Menlo Park, CA 94025

COMPUTERTOWN, USA

Computer enthusiasts in any town can copy the program in Menlo Park, Computertown, USA, by taking computers to public libraries, pizza parlors, churches or any other places where people get together and share experiences. Adults will get a chance to learn about home computers and children will have the opportunity to learn how to use, enjoy and program.

For more information send a stamped, self-addressed envelope to Computertown, USA, P.O. Box 310, Menlo Park, CA 94025.□

The Pascal Notebook

Chapter 9

Final Chapter

By Henry Davis, Associate Editor

Throughout this series you have seen innumerable references to P-code and P-machine. Implementing the P-machine is really the crux of obtaining Pascal for any machine. Table 1 is a summary of the individual instructions and standard procedures of the P-machine. Each instruction corresponds to a subroutine that must be implemented in assembly language. In terms of size, the programming task requires about six thousand lines of code. Fortunately, much of this code is readily available or is simple to implement. The major problem is that it is tedious work.

Many different companies offer Pascal for most of the popular microcomputers. Prices range from a low of $250 to over $2500 per copy. Depending on your programming expertise and desires, a purchased system can be your best bet. Regardless of whether you purchase or implement Pascal, a good library of Pascal references is a must. Not only do they teach you Pascal but also many algorithms are detailed. Table 2 lists a few of the available Pascal books.

Once you have an interpretive Pascal running, what next? The first idea that comes into most people's mind is to produce native code for their particular computer. The motivation is to decrease the execution time because interpreters are relatively slow. In theory, producing native code is faster because there is no software overhead associated with execution of an instruction. Unfortunately, any instruction set is less than optimal for any practical problem because there is always unnecessary generality or a lack of generality associated with the instructions. Interpreters circumvent this problem by providing a tailored instruction set. This results in shorter programs but longer execution time.

Threaded code provides a compromise between native code and interpretation. Typical compiled threaded code runs 2-5% slower and is 10-20% shorter than its equivalent native code. Additionally, it is still relatively easy to transport to new machines.

A threaded code computer would work in the following manner:

Step 1 fetch S — the PCTH word in memory
Step 2 a) execute the routine at location S
b) increment PC by 1
Step 3 go to Step 1

This scheme can be implemented very easily in software. If we let the PC be a general register of the 9900 microprocessor then each subroutine specified in Step 2a is ended with the instructions

```
MOV     *PC+, PCTEMP
B       *PCTEMP
```

This links each routine together in the order necessary to execute any given program. These two instructions perform exactly Steps 1 and 2 of the algorithm. What has really happened is that an interpretive code is used but without an interpreter. The technique of threaded code was discovered by James Bell of Digital Equipment Corporation and has been used extensively by DEC in their compilers. In addition to being faster than an interpreter, threaded code only contains the necessary service routines rather than all service routines.

Now that you have seen a little of Pascal you should be able to see that it can be a powerful programming tool when used properly. Investigation of the concepts introduced in this series will lead you through many fascinating areas of computer science. I hope that the materials presented will make your programming easier and more enjoyable.

Table 1. Symbolic Instructions of PASCAL-CODE

For the sample interpreter, each instruction is packed into a 30-bit field. The op-code occupies a 7-bit field, parameter P a 4-bit field, and parameter Q a 19-bit (address) field. For the relational instructions the C parameter is put into the P field; in all other cases the C parameter is used to modify the instruction code. Sometimes, the Q field may be symbolic. This is indicated by an asterisk (*) in the description below.

Alphabetic List of Instructions:

code	mnemonic	parameters	description
40	ABI		absolute value of integer
41	ABR		absolute value of real number
28	ADI		integer addition
29	ADR		real addition
43	AND		Boolean "and"
26	CHK	C P Q	check against upper and lower bounds
60	CHR		convert integer to character
15	CSP	Q	call standard procedure
12	CUP	P Q	call user procedure*
57	DEC	C Q	decrement
45	DIF		set difference
53	DVI		integer division

54	DVR				real division	
13	ENT		P	Q	enter block*	
27	EDF				test on end of file	
17	EQU	C		(Q)	compare on equal	
24	FJP			Q	false jump*	
34	FLO				float next to the top	
33	FTL				float top of the stack	
19	GEQ	C		(Q)	greater or equal	
20	GRT	C		(Q)		
10	INC	C		Q	increment	
9	IND	C		Q	indexed fetch	
48	INN				test set membership (in)	
46	INT				set intersection	
44	IOR				Boolean, "inclusive or"	
16	IXA			Q	compute indexed address	
5	LAO			Q	load base-level address	Assembler may generate the same instruction
56	LCA			Q	load address of constant	
4	LDA		P	Q	load address with level P	
7	LDA	C		Q	load constant	
1	LDO	C		Q	load contents of base-level address	
21	LEQ	C		(Q)	less than or equal	
22	LES	C		(Q)	less than	
0	LOD	C	P	Q	load contents of address	
49	MOD				modulus	
55	MOV			Q	move	
51	MPI				integer multiplication	
52	MPR				real multiplication	
11	MST		P		mark stack	
18	NEQ	C		(Q)	not equal	
36	NGI				integer sign inversion	
37	NGR				real sign inversion	
42	NOT				Boolean "not"	
50	ODD				test on odd	
59	ORD	C			convert to integer	
14	RET	C			return from block	
30	SBI				integer subtraction	
31	SBR				real subtraction	
32	SGS				generate singleton set	
38	SQI				square integer	
39	SQR				square real	
3	SRO	C		Q	store at base level address	
6	STO	C			store indirect	
58	STP				stop	
2	STR	C	P	Q	store at level P	
35	TRC				truncation	
23	UJP			Q	unconditional jump*	
47	UNI				set union	
25	XJP			Q	indexed jump*	
8			P	Q	load constant indirect, an assembler-generated instruction	
61	UJC				error in case statement	

Type of the operands on the top of the stack for the instructions (the first element corresponds to the top of the stack)

before	after	the instruction
int	int	ABI, NGI, SQI
real	real	ABR, NGR, SOR
bool	bool	NOT
adr	adr	DEC, INC
int	real	FLT
int	bool	ODD
int	set	SGS
real	int	TRC
adr	bool	EOF
int	↓	
int	int	ADI, DVI, MOD, MPI, SBI
real	↓	
real	real	ADR, DVR, MPR, SBR
bool	↓	
bool	bool	AND, IOR
int	↓	
adr	adr	IXA
set	↓	
set	set	DIF, INT, UNI
1	any*	LOD
1	adr	LAO, LCA, LDA
1	any*	LDC, LDO
any	↓	SRO, STR
bool	↓	FJP
any*	↓	EQU, GEQ, GRT, LEQ, LES, NEQ
any*	bool	
any*	↓	
adr	↓	STO
no action on top of stack		CHK, WP, STR, UJP, UJC
special		CSP, ENT, MST, RET
adr	any*	IND
set	↓	
int	bool	INN
real	real	
int	real	FLO
adr	↓	
adr	↓	MOV
int	char	CHR
any	int	ORD

*the type is given by the C-parameter of the instruction

Mnemonics of Standard Procedures/Functions of the Hypothetical Stack Computer

The only argument of the CSP (*C*all *S*tandard *P*rocedure) instruction is of mnemonic representing a Standard Procedure/Function. The integer representation or code of this mnemonic is put into the address part of an instruction by the assembler.

Alphabetic List of Standard Procedures/Functions:

Mnemonic	Code	Description
ATN	19	Computes the *arctan* function for the value on the top of the stack and leaves the value of the result on the top of the stack.
COS	15	Computes the *cosine* function for the value on the top of the stack and leaves the value of the result on top of the stack.
LLN	2	Checks the EOLN condition for the file specified on the top of the stack; the result of this check is left on the top of the stack.
EXP	16	Computes the function e^y where y is the value on the top of the stack; the result is left on the top of the stack.
GET	0	Performs *get* on the file specified by the top of the stack and appropriately fills the buffer associated with it.
LOG	17	The natural logarithm is computed for the value on the top of the stack; the computed value is left on the top of the stack.
NEW	4	The top of the stack specifies the size of element to be allocated from the free storage; the address of the element is to be stored in the pointer variable whose address is to be found below the top of the stack.
PUT	1	Performs the put operation on the file specified by the top of the stack; the buffer is now initialized to 'undefined.'
RDC	13	Reads a character from the file specified on the top of the stack and assigns it to the variable whose address is below the top of the stack; note the automatic updating of the buffer associated with the file.
RDI	11	Reads an integer from the file specified on the top of the stack and assigns it to a variable whose address is below the top of the stack; note the automatic updating of the buffer associated with the file.
RDR	12	Reads a real number from the file specified on the top of the stack and assigns it to a variable whose address is below the top of the stack; note the automatic updating of the buffer associated with the file.
RLN	20	The top of the stack specifies a file on which a READLN is performed; note the automatic updating of the buffer associated with the file.
RST	2	Sets the 'new' pointer (heap pointer) to the pointer value on the top of the stack.
SAV	20	Saves the current value of the 'new' pointer (heap pointer) at the address specified on the top of the stack.
SIN	14	Computes the *sine* function for the value on the top of the stack; the result is left on the top of the stack.
SQT	18	The *square root* of the value on the top of the stack is computed and the result is left on the top of the stack.
WLN	7	Performs WRITELN on the file specified on the top of the stack.

WRC	10	Writes on the file specified on the top of the stack a character which is found immediately below the element below the top of the stack. Just below the top is the number of characters to be written out.*
WRI	8	Writes on the file specified by the top of the stack an integer whose value is given immediately below the element below the top of the stack. Just below the top is the number of characters to be written out.*
WRR	9	Writes on the file specified by the top of the stack a real number whose value is given immediately below the element below the top of the stack. Just below the top is the number of characters to be written out.*
WRS	6	Writes on the file specified by the top of the stack a string of characters; immediately below the top of the stack is the value of the actual length of the string; below this is specified the actual length to be written out — if the actual length is less than the number of characters to be written out, then sufficient number of initial blank characters are written out; below the number of characters to be written out is an address where the actual string can be found.

*if necessary leading blanks are filled in.

As is standard with all stack machines, the use of operands by the above standard procedures/functions causes them to be removed from the top of the stack. In the use of functions, the result is pushed onto the stack.

Type of operands on the top of the stack for the standard procedures and functions (the first element corresponds to the top of the stack)

real	real	ATN, COS, EXP, LOG, SIN, SQT
adr	bool	ELN
adr	↓	GET, PUT, RLN, SAV, RST, WLN
adr	↓	RDC, RDI, RDR
adr	↓	
int	↓	NEW
adr	↓	
adr	↓	WRI
int	↓	
int	↓	
adr	↓	
int	↓	WRR
real	↓	
adr	↓	
int	↓	WRS
int	↓	
adr	↓	
adr	↓	WRC
int	↓	
char	↓	

Table 2.

Bowles, K.L., *Microcomputer Problem Solving Using Pascal,* Springer-Verlag, New York, 1977.

Bowles, K.L., Introduction to Computer Science, Springer-Verlag, New York, 1977.

Conway, R., Gries, D., and Zimmerman, C., *A Primer of Pascal,* Winthrop Publishers, Boston, 1976.

Conway, R., Gries, D., and Zimmerman, C., *A Primer on Structured Programming Using Pascal,* Winthrop Publishers, Boston, 1976.

Grogono, Peter, *Programming in Pascal,* Addison-Wesley, Reading, MA, 1978.

Jensen, K., and Wirth, N., *Pascal-Users Manual and Report,* Springer-Verlag, New York, 1974.

Nori, K.V., Ammann, U., Jensen, K., Nageli, H.H. and Jacobi, Ch., *The Pascal (P) Compiler: Implementation Notes,* 1974.

Schneider, C.M., Weingart, S., and Perlamm, D.M., *Introduction to Problem Solving and Programming with Pascal,* Wiley, New York, 1977.

Wirth, N., *Algorithms + Data Structures = Programs,* Prentice-Hall, Englewood Cliffs, NJ, 1976.

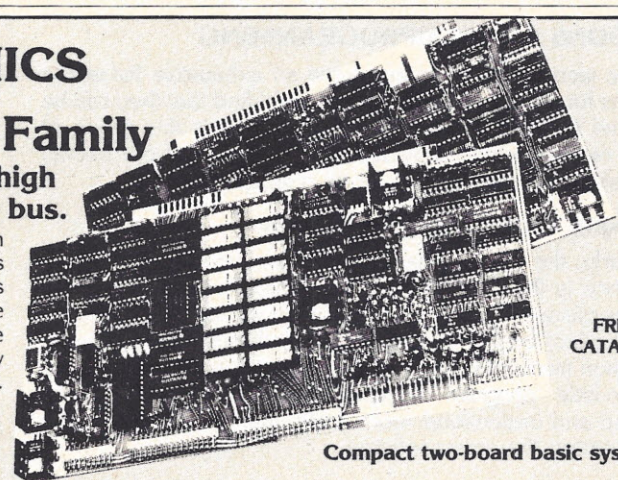
Wirth, N., *Systematic Programming — An Introduction,* Prentice-Hall, Englewood Cliffs, NJ, 1972.

□

Complete COLOR GRAPHICS With The CAT-100 Family

The original 256-color imaging system with high resolution video frame grabber for the S-100 bus.

Its digitizer captures a video frame in 1/60th of a second and stores it in its on-board image memory, with software selectable resolutions ranging from 256 to 1280 pixels per TV line. The image memory is accessible through a 2K window in the S-100 address space. The compact two-board basic system digitizes and displays 480x512 single bit pixels or 240x256 four bit pixels. The video generator displays 16 gray levels or 16 colors on standard B&W, NTSC or RGB color TV monitors.



FREE CATALOG

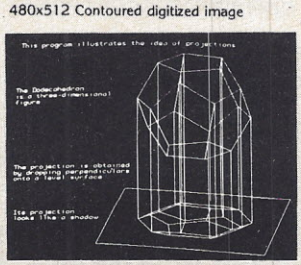
Compact two-board basic system



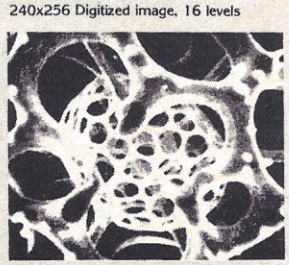
480x512 Contoured digitized image



240x256 Digitized image, 16 levels



480x512 Computer-generated



240x256 Digitized image, 16 levels

Features:

- Input capability from RS-170 and other TV signals
- Variety of synchronization choices
- 2 selectable video A/D conversion circuits
- Choice of 1, 2 or 4 bits per pixel
- 32K-byte image memory on the basic unit
- Expandable to 256K bytes with 24-bit color map
- Lightpen input
- Photographic trigger control input
- Character generator
- Software selectable system parameters
- Interfaces for LSI/11, TRS-80 and other processors

Accessories:

- NTSC and RGB Sony Trinitron color monitor
- High sensitivity, high resolution lightpen
- Hard copy electrostatic printer/plotter with interface
- Large variety of software packages

ROBOTICS-AUTOMATED COUNTING AND MEASURING-QUALITY CONTROL MONITORING-PATTERN RECOGNITION-IMAGE PROCESSING-MEDICAL AND SCIENTIFIC RESEARCH

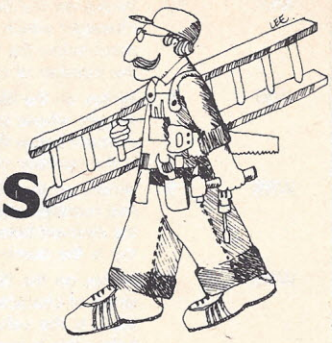
DIGITAL VIDEO SYSTEMS

441 California Ave.,
Palo Alto, CA 94306
415/494-6088

Dept. 16

Using and Building Micro-Based Systems

Chapter Four



By David Marca, Associate Editor

INTRODUCTION

The concept of a Subprogram is not new to computer languages. Early on, the need for such a language feature was perceived. FORTRAN I (back in 1957) had no language statements to support subprogramming. Shortly thereafter, however, FORTRAN II (in 1958) appeared with several big improvements — one of which was the ability to create subprograms.[1] It is somewhat surprising that the syntax allowing definition and use of FORTRAN subprograms has remained unchanged through two additional generations of the language.

Since this is such an important part of the language, this entire chapter is dedicated to the subject. We will investigate some major reasons why subprograms are important, the types and capabilities of FORTRAN subprograms, and see how to define and properly use subprograms in the FORTRAN environment. While there was a general understanding of a subprogram's usefulness in the early days of FORTRAN, most of the concepts and practical guidelines presented in this article were either not known or not well-understood at that time. The concepts and practicalities presented represent the experience gained by using subprograms for some two decades by many professionals.

REASONS FOR SUBPROGRAMMING

If we were to sit down and write an exhaustive list of the reasons for subprogramming, we would find that they can be grouped into several general categories. It is this group of major reasons which provides the foundation of our overall understanding, and each is worth a separate look.

Breaking Up a Problem's Solution

Clearly, the most fundamental reason for a subprogram's existence is its ability to split a solution to a problem into smaller pieces. The programmer is given the power to arbitrarily collect lines of code, and physically separate each such collection from one another. Each collection really turns into its own little "program" which is a part of the whole, and can be read and understood more easily than a single program representing the entire solution.

Isolate Regularities and Separate Differences

While the programmer does have the option to arbitrarily collect and separate lines of code, some guidelines must be followed. Clearly, the things we want to collect together are the lines of code which do the same job in numerous areas of the problem's solution: *Separating code into appropriate modules is an important aspect of structuring a program. . . the subroutine call permits us to summarize the* irregularities *in the argument list, where we can see quickly what is going on. The subroutine itself summarizes the* regularities *of the code, so patterns need not be used.*[2]

The contexts within which the subprogram is used reflects the differences in the appropriate parts of the problem's solution. This isolation of regularities and separation of differences apply equally well to functions (lines of code) and data (variables) used by the funcitons. The subprogram provides an area where a particular piece of data can "hide" from the rest of the subprograms.[3]

Replaceable/Reusable Pieces

When similarities are isolated properly, a subprogram can be "plugged in" very easily to other areas of the same (or even different) software system. This feature also works in reverse, because a particular subprogram can be "pulled out" and replaced. Replacing generally occurs when a subprogram can no longer operate properly due to a change in the application environment (e.g., change in business procedures), or when the subprogram has been poorly written and isn't worth rewriting.

Better Portability of Software

An important concept built from the combination of good isolation principles and the "plug in-pull out" notion is that of portability (the ease of conversion of a [software] system from one environment to another).

This occurs when a software system is written so as to isolate all areas of machine *dependent* logic into a group of subprograms. Then, only those subprograms containing machine dependent logic need to be rewritten in order to run the software on a new machine. Subprogramming gives the programmer greater control in defining those areas of machine dependence.

Easier Testing and Debugging

Along with the capability of building separate pieces goes the ability to test each piece separately. Thus the programmer can test each piece before integrating it into the whole. There are several ways to do this testing and integrating (top-down and bottom-up are the most common integration methods; phased and incremental are the most common methods of testing[4]), and this deserves discussion far beyond the scope of this article. All of the methods do, however, rely upon the fact that only a small amount of test data is needed to adequately check out a subprogram's operation versus the amount of test data required for a very large program. Clearly, the less code there is, the easier it is to test.

In the same vein, when errors are detected in a software system built using subprograms, they can be tracked down more easily. Hence, this reduces the time needed to find and fix an error once it has occurred. Software systems comprised of one extremely large program, void of subprograms, are usually a programmer's nightmare when an error is discovered and must be corrected. Easier testing and debugging is certainly a blessing in the microcomputing environment because sophisticated tools are not available as they are in large mainframe computer systems.

Aid Towards Writing Correct Programs

The contribution subprograms make in simplifying testing and debugging is significant. However, testing a program does not ensure a program is correct.

Showing that a program is correct solely by testing is quite impractical given today's machines and programming languages. Test data — no matter how large and complex — is usually a very small sample of possible program input. To rigorously show program correctness by testing would require us to run the program on all possible input data. Clearly, that is an absurd undertaking.[5]

Programmers do, however, have a much better chance of writing correct programs when the program size is small. By using subprograms to create small portions of an entire software system, the chances of each individual unit being correct are greater than the chance that a *single* program, representing the same software system, is correct.

CHARACTERISTICS OF FORTRAN SUBPROGRAMS

Now that a context for subprogramming in general has been established, let's investigate the overall characteristics of just FORTRAN subprograms. The general principles that will be discussed apply to all FORTRAN subprograms, regardless of a subprogram's type. These principles are presented here as a group not only to provide a technical foundation, but also to alleviate redundancy when describing each different type of subprogram. Each characteristic is discussed separately.

Defined Boundary Around FORTRAN Statements

By their very nature, FORTRAN subprograms separate groups of FORTRAN statements from one another. The statements are grouped together so that the entire collection performs a single function or multiple functions depending upon subprogram type and content. Each FORTRAN subprogram has its own syntax for defining the boundary of the collection.

Defined Interface Point

An "interface point" is the place where the subprogram hooks up with the rest of the software system. The name of the subprogram is the interface point. It is across this point of connection that data is usually given to the subprogram and given back by the subprogram. The entire interface is formally defined by the name and any data that is given to and/or returned by the subprogram. The entire interface is defined in the first FORTRAN statement of the subprogram.

Consistent and Limited Transfer of Control

A "transfer of control" occurs when a FORTRAN program (or subprogram) "stops" the execution of its statement and allows a subprogram to start executing. This process is termed "calling the subprogram." A transfer also occurs when a subprogram is finished executing its statements and tells the program or subprogram which originally "stopped" to "resume" its execution. This process is termed "returning to the caller." The interface point is the only place where these transfers of control are performed. Hence, these transfers are limited because execution "stops" immediately before the interface point, and execution "resumes" immediately after the interface point (Figure 1).

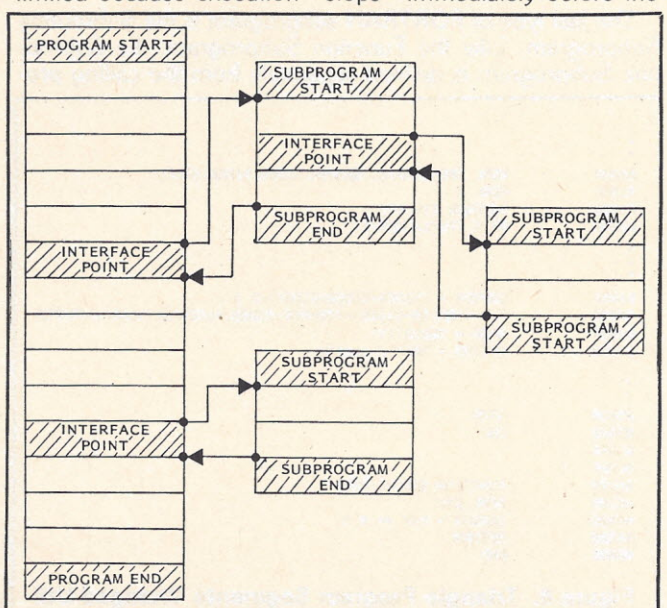


Figure 1. Example of Transfers of Control

Limited Scope of Variables

Variables can be defined within a subprogram just as they can be defined in a regular program. These variables are, however, usually known only to the subprogram in which they are defined. This restrictive knowledge about a variable is known as the *scope* of a variable.

The scope rules for FORTRAN are very simple as compared to languages like PL/I and Pascal. A variable defined in a subprogram is *exclusively* owned by that subprogram unless a COMMON statement appears after the definition. The COMMON statement allows the variable to be known by all subprograms with the same defining statement and COMMON statement (Figure 2). Thus a variable is either known by only one subprogram or is (potentially) known by all subprograms.

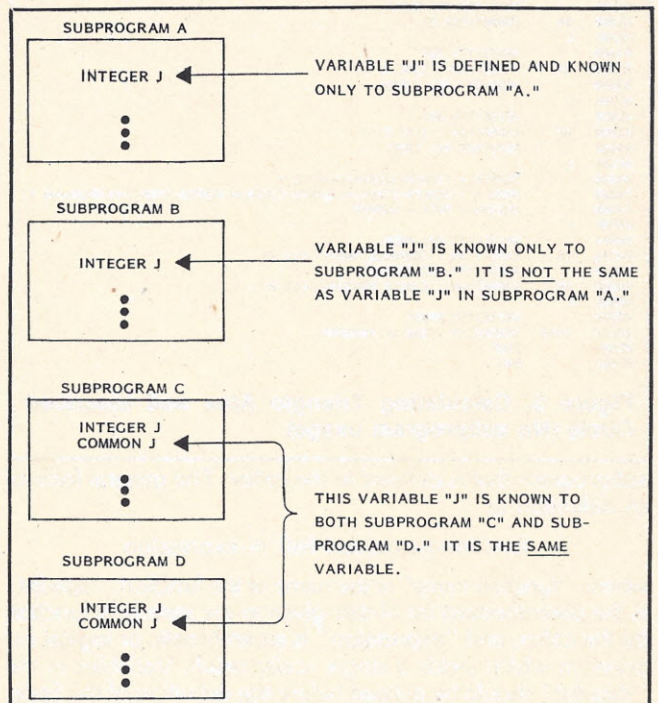


Figure 2. An Example of FORTRAN Scope Rules

Separately Compilable and Linkable

Since the subprogram concept separates groups of FORTRAN code, subprograms are usually kept as separate source files. These files can then be individually compiled and linked when necessary. Keeping subprograms physically separate in this way allows easy reuse of any compiled subprogram. No additional development time is needed — just call it in the right places, and link the appropriate REL file when creating the executable OBJECT file.

TYPES OF FORTRAN SUBPROGRAMS

Using the general subprogram characteristics as a foundation, we will now expand those principles as they apply to each type of FORTRAN subprogram. There are four general types of FORTRAN subprograms: Statement Functions, Library Functions, Function Subprograms, and Subroutine Subprograms. Each will be discussed separately. Throughout this investigation, we will be taking an already working program (Figure 3) and modifying it to utilize each of the different subprogram types.

The first three types of FORTRAN subprograms all fall into the function category. In FORTRAN, a function is called (and usually given data) and returns a *single scalar* result to the caller. The Statement Function is the simplest form of a function. It is a very special subprogram because it is the only

```
00100   C-------------------------------------------
00200   C  NAME: TRIANGLE
00300   C  PURPOSE: CALCULATE AREA OF TRIANGLE AND
00400   C           RADIUS OF INSCRIBED CIRCLE.
00500   C  DESCRIPTION:
00600   C       THREE SIDES OF TRIANGLE ARE ENTERED.
00700   C       SEMI-PERIMETER OF TRIANGLE FIRST CALCULATED.
00800   C       AREA OF TRIANGLE CALCULATED USING SEMI-PERIMETER VALUE.
00900   C       RADIUS OF CIRCLE CALCULATED USING SEMI-PERIMETER VALUE.
01000   C       AREA AND RADIUS VALUES ARE PRINTED.
01100   C  DEVICES:
01200   C       TTY - CONSOLE TELETYPEWRITER
01300   C       KEY - KEYBOARD
01400   C  NOTES: DOES NOT CHECK FOR BAD INPUT
01500   C-------------------------------------------
01600   C
01700         REAL AREA,RADIUS,SEMPER,SIDEA,SIDEB,SIDEC
01800         INTEGER TTY,KEY
01900         DATA TTY/1/,KEY/1/
02000   C
02100   C     . . . . . .
02200   C
02300         WRITE(TTY,10)
02400   10    FORMAT(1H ,'START OF PROGRAM:')
02500   C
02600         WRITE(TTY,20)
02700   20    FORMAT(1H ,'PLEASE ENSURE INPUT DATA HAS CONSTANT UNITS.')
02800   C
02900         WRITE(TTY,30)
03000   30    FORMAT(1H ,'SIDE A:')
03100         READ(KEY,40) SIDEA
03200   40    FORMAT(F10.2)
03300   C
03400         WRITE(TTY,50)
03500   50    FORMAT(1H ,'SIDE B:')
03600         READ(KEY,40) SIDEB
03700   C
03800         WRITE(TTY,60)
03900   60    FORMAT(1H ,'SIDE C:')
04000         READ(KEY,40) SIDEC
04100   C
04200         SEMPER = (SIDEA+SIDEB+SIDEC)/2.0
04300         AREA = ((SEMPER-SIDEA)*(SEMPER-SIDEB)*(SEMPER-SIDEC)*SEMPER)**0.5
04400         RADIUS = AREA / SEMPER
04500   C
04600         WRITE(TTY,70) AREA
04700   70    FORMAT(1H ,'TRIANGLE AREA=',F15.2)
04800         WRITE(TTY,80) RADIUS
04900   80    FORMAT(1H ,'CIRCLE RADIUS=',F15.2)
05000   C
05100         WRITE(TTY,9999)
05200   9999  FORMAT(1H ,'END OF PROGRAM:')
05300         STOP
05400         END
```

Figure 3. Calculating Triangle Area and Inscribed Circle (No subprogram usage)

subprogram that is *defined in the caller.* The general form of its definition is:

function-name(date-list) = expression

where: "function-name" is the name of the function, "data-list" is the parenthesized list of data given to the statement function by the caller, and "expression" is an arithmetic or logical expression which yields a single scalar result. Variables in the "data-list" should be defined before the definition of the Statement Function. The statement function should be defined after all other variables, and before any executable statements. The first in our set of examples shows the use of a statement function in calculating the square root (Figure 4).

```
      .
      .
      .
01800         REAL AREA,RADIUS,SEMPER,SIDEA,SIDEB,SIDEC
01900         REAL X
02000         INTEGER TTY,KEY
02100         DATA TTY/1/,KEY/1/
02200   C
02300         SQROOT(X) = X ** 0.5
02400   C
02500   C     . . . . . .
02600   C
      .
      .
      .
04600         SEMPER = (SIDEA+SIDEB+SIDEC)/2.0
04700         X = (SEMPER-SIDEA)*(SEMPER-SIDEB)*(SEMPER-SIDEC)*SEMPER
04800         AREA = SQROOT(X)
04900         RADIUS = AREA / SEMPER
      .
      .
      .
05800         STOP
05900         END
```

Figure 4. Triangle Program Segments Changed Utilizing a Statement Function.

The second type of FORTRAN function is the Library Function. As its name implies, a Library Function is one of several functions that reside in a separate library for general use by all programmers. The Library Function has been previously defined, and is part of the FORTRAN operating environment. All the programmer needs to do is use the Library Function (by giving the correct name and data required) in the source code. The FORTRAN compiler will properly translate the program, and the link/loader will properly incorporate an executable copy of the Library Function into the OBJECT file. Our next example shows the use of a square root library function instead of the Statement Function (Figure 5).

```
      .
      .
      .
01800         REAL AREA,RADIUS,SEMPER,SIDEA,SIDEB,SIDEC
01900         REAL X
02000         INTEGER TTY,KEY
02100         DATA TTY/1/,KEY/1/
      .
      .
      .
04400         SEMPER = (SIDEA+SIDEB+SIDEC)/2.0
04500         X = (SEMPER-SIDEA)*(SEMPER-SIDEB)*(SEMPER-SIDEC)*SEMPER
04600         AREA = SQRT(X)
04700         RADIUS = AREA / SEMPER
      .
      .
      .
05600         STOP
05700         END
```

Figure 5. Triangle Program Segments Changed Utilizing a Library Function.

The third type of FORTRAN function, the Function Subprogram, is very similar to the Library Function type. The only difference is that the programmer must actually write FORTRAN statements comprising the function. The FUNCTION statement and END statement are used to define the functions boundary. The FUNCTION statement is also used to define the interface point in the following way:

FUNCTION function-name (data-list)

and must be the first FORTRAN statement in the subprogram.

Once the boundary and interface points are established, the programmer should define all variables in the "data-list" and then any other variables used by the function. After defining all data, the programmer can then proceed to writing the FORTRAN statements which do the actual work. In order to give a result back to the caller, the programmer must establish the return value by the following:

function-name = expression

where "function-name" is the *same* name used to define the Function Subprogram, and "expression" is an arithmetic or logical expression which will yield the desired result of the function. Finally, transfer of control is passed back to the caller by the use of the RETURN statement. Figure 6 shows a caller and a called Function Subprogram.

The last type of FORTRAN subprogram is the Subroutine Subprogram. Like the Function Subprogram, the Subroutine Subprogram is defined separately from the calling pro-

```
      .
      .
      .
01800         REAL AREA,RADIUS,SEMPER,SIDEA,SIDEB,SIDEC
01900         REAL X
02000         INTEGER TTY,KEY
02100         DATA TTY/1/,KEY/1/
      .
      .
      .
04600         SEMPER = (SIDEA+SIDEB+SIDEC)/2.0
04700         X = (SEMPER-SIDEA)*(SEMPER-SIDEB)*(SEMPER-SIDEC)*SEMPER
04800         AREA = SQROOT(X)
04900         RADIUS = AREA / SEMPER
      .
      .
      .
05800         STOP
05900         END
06000   C
06100   C
06200         FUNCTION SQROOT(EXP)
06300         REAL EXP
06400         SQROOT = EXP ** 0.5
06500         RETURN
06600         END
```

Figure 6. Triangle Program Segments Changed Utilizing a Function Subprogram.

gram or subprogram. Similarly, it has a boundary defined by the SUBROUTINE and END statements, and an interface point defined by:

SUBROUTINE subroutine-name (data-list)

and transfers control back to the caller using the RETURN statement. The major difference stems from the fact that a Function Subprogram can return only *one* result to the caller, while the Subroutine Subprogram can return many results to the caller. Returning a value to the caller is accomplished by assigning the result of an expression to one of the variables in the "data-list."

Our final example (Figure 7) shows the first variable in the "data-list" being used for input, and the second variable being used for output of the square root computation.

```
01800          REAL AREA,RADIUS,SEMPER,SIDEA,SIDEB,SIDEC
01900          REAL X
02000          INTEGER TTY,KEY
02100          DATA TTY/1/,KEY/1/
04600          SEMPER = (SIDEA+SIDEB+SIDEC)/2.0
04700          X = (SEMPER-SIDEA)*(SEMPER-SIDEB)*(SEMPER-SIDEC)*SEMPER
04800          CALL SQROOT(X,AREA)
04900          RADIUS = AREA / SEMPER
05800          STOP
05900          END
06000    C
06100    C
06200          SUBROUTINE SQROOT(EXP,ROOT)
06300          REAL EXP,AREA
06400          ROOT = EXP ** 0.5
06500          RETURN
06600          END
```

Figure 7. Triangle Program Segments Changed Utilizing a Subroutine Subprogram.

GENERAL COMMUNICATION PROTOCOL

Our current understanding of FORTRAN subprogram concepts, characteristics, and types will allow us to investigate FORTRAN subprogram communication details. The principles discussed in this section apply in general to all FORTRAN subprograms, but have special value in regard to subroutines. So our discussion will provide emphasis on SUBROUTINE aspects of communication.

Subroutines communicate with their calling environment through what we have termed as the "data-list." The caller of the subroutines and the subroutine itself have different *viewpoints* with regard to the "data-list." The caller uses the "data-list" for an entirely different purpose than does the subroutine. Thus, a distinction is made between the caller and the subroutine views of the "data-list."

In the caller view, values are given to some of the members of the "data-list," and values are taken from other members of the "data-list." These "members" will be termed *arguments*. From the subroutine view, "members" of the "data-list" are used inside the subroutine to do its job and to return results. They will be termed *parameters*. A list of these "members" will be termed *argument list* and *parameter list* for the caller and subroutine respectively.

The way in which arguments are given to a subroutine is through some sort of communication region. This region is usually different in layout for each implementation of FORTRAN, however, a good generic model of the communication region is shown in Figure 8. As shown, there are *implicit* arguments and *actual* arguments. The actual arguments represent those arguments which are given to the subroutine by the caller. Implicit arguments represent data needed by the FORTRAN run-time system, and which is used to properly execute the subroutine and return. Examples of implicit arguments are the return address, and the number of actual arguments.

It is important to note that the actual arguments in the communication region are usually *addresses* of the variables

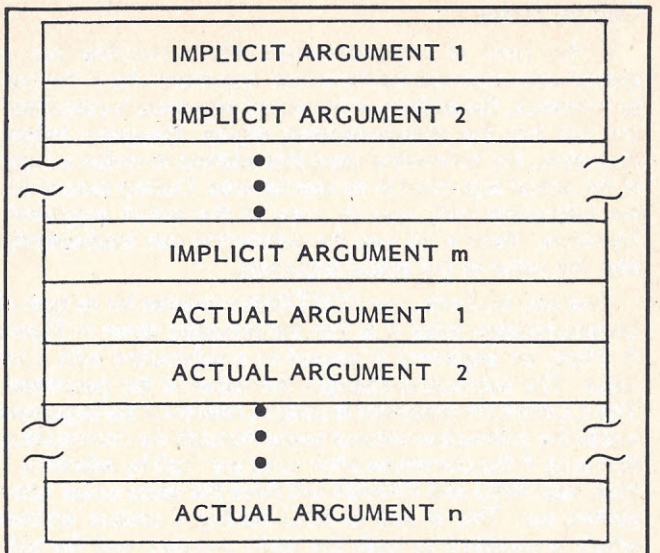


Figure 8. Subroutine Communication Region Layout

passed. FORTRAN does not usually put a variable in the communication region, but instead uses the address to *reference* the variable. The rules for using these addresses can differ from one implementation of FORTRAN to the next. We will discuss two separate sets of rules relevant to FORTRAN compilers. These rules are explained here in terms of a single argument, but are actually applied to each argument in the list.

Call by Reference

This is the easiest type of communication protocol to implement inside a compiler, and hence this is the most frequent type of implementation found (especially in microcomputer environments with limited memory resources). Prior to calling the subroutine, the address of an actual argument is calculated: a) if it is an expression, its result is computed, stored in a temporary, and the address of the temporary is used, b) if it is a variable or constant, the address of the variable or constant is used. This address is then put into the appropriate slot in the communication region for that specific actual parameter. The subroutine uses the address in the communication region to retrieve and/or set values accordingly. A big danger to this type of implementation lies in the fact that the subroutine can alter the value of the actual parameter at any time.

```
00100          INTEGER ARG
00200          COMMON /REGION/ARG
00300          ARG = 5
00400          CALL CHANGE(ARG)
00500          STOP
00600          END
00700    C
00800    C
00900          SUBROUTINE CHANGE(PARAM)
01000          INTEGER PARAM
01100          INTEGER TTY
01200          INTEGER ARG
01300          COMMON /REGION/ARG
01400          TTY = 1
01500          PARAM = PARAM + 1000
01600          WRITE(TTY,10) PARAM,ARG
01700    10    FORMAT(1H ,'PARAM=',I4,'  ARG=',I4)
01800          RETURN
01900          END
```

Figure 9. Program to Check Out FORTRAN Communication Rules

Call by Value

In this form of communication, the *subroutine* has a special area reserved for the actual argument. As in the call by reference, the address of the actual argument is calculated and put into the communication region. However, before execution, the subroutine uses this address to make a copy of the actual argument in its special area. During execution, the subroutine only uses its copy of the actual argument. Therefore, there is no way the subroutine can inadvertently alter the value of the actual argument.

One way to check your FORTRAN compiler for its type of communication rules is to run the program given in Figure 9. Here, an argument is passed to a subroutine with a set value. The subroutine changes the value of the parameter. The COMMON statement is used to reference the argument inside the subroutine without going through the communication area. If the communication rules are "call by reference," then both ARG and PARAM will have the *same* value when written out. This is due to ARG having a unique address which is modified by both the main program and the subroutine. If the communication rules are "call by value," then ARG and PARAM will have *different* values when written out. This is due to the subroutine making a copy of ARG's value in its own special area.

PRACTICAL GUIDELINES

There are good ways to use subprograms, and there are bad ways to use subprograms when creating a software system. Throughout the years, experience with using subprograms has evolved many rules. Some of these rules relate to the style of writing programs[6,7] and some rules have been generalized into principles which can be applied to programming and even design.[8]

The extent to which the guidelines presented here are followed is due not only to their potential benefit to the resulting software system, but also due to the environment in which the software system is built and used. Subprograms give you, the programmer, the ability to do many things. However, it is the proper use of their capabilities that makes a good software system.

Use a Template to Start Building a Subprogram

Last month we saw the usefulness of a template as a shell to create programs. We can again use this principle by defining a block comment area with additional entries for calling sequence and argument descriptions (Figure 10). Also included in the template are the boundary statements, which give the subprogram's overall structure. Since the interface varies greatly, the template should not have any predefined statements for the interface.

The template is not only useful as a shell for a subprogram, but is also the perfect vehicle for establishing consistent debugging logic. The most common form is the trace message (Figure 11). This example shows the use of the COMMON statement to pass in the trace device code and a trace on/off switch. If a trace is desired, the appropriate messages are printed. In order to properly use this trace facility, the main program must also define the trace device and trace switch as shown, and set values appropriately.

```
00100   C-------------------------------------
00200   C  NAME:
00300   C  PURPOSE:
00400   C  CALLING SEQUENCE:
00500   C  INPUT ARGUMENTS:
00600   C  OUTPUT ARGUMENTS:
00700   C  DESCRIPTION:
00800   C  DEVICES:
00900   C  NOTES:
01000   C-------------------------------------
01100   C
01200          SUBROUTINE *
01300   C
01400   C  .  .  .  .  .  .  .  .  .  .  .  .
01500   C
01600          RETURN
01700          END
```

Figure 10. Subroutine Template Layout

Communicate as Much as Possible Through the Interface Point

Any data used and/or returned should be explicitly defined in the interface to the subprogram. In this way, the caller is dependent only upon the values it gives the subprogram and the values returned by the subprogram. The subprogram, in turn, is dependent upon only those variables in the parameter list in order to do its job. This form of subprogram communication gives the total software system a high degree of being a "good" system.[9]

```
00100   C-------------------------------------
00200   C  NAME:
00300   C  PURPOSE:
00400   C  CALLING SEQUENCE:
00500   C  INPUT ARGUMENTS:
00600   C  OUTPUT ARGUMENTS:
00700   C  DESCRIPTION:
00800   C  DEVICES:
00900   C  NOTES:
01000   C-------------------------------------
01100   C
01200          SUBROUTINE *
01300          LOGICAL TRACE
01400          INTEGER TRCDEV
01500          COMMON /TRCCOM/TRACE, TRCDEV
01600   C
01700   C  .  .  .  .  .  .  .  .  .  .  .  .
01800   C
01900          IF (TRACE) WRITE(10, TRCDEV)
02000    10    FORMAT(1H , 'START *')
02100   C
02200          IF (TRACE) WRITE(9999, TRCDEV)
02300    9999  FORMAT(1H , 'END   *')
02400          RETURN
02500          END
```

Figure 11. Subroutine Template with Trace Facilities

Remember that there is a practical limit to the above guideline. A good example is the subroutine template in Figure 11. There are three good reasons why the trace variables are not communicated through the interface point:

1. they appear in *all* subprograms
2. they clutter up the interface
3. they are not part of the subroutine's actual job.

Clearly, the above are grounds for communication through the COMMON region.

Keep Parameter Lists Small

Clearly a list of three variables is easier to read and understand versus a list of 20 variables. We should strive to keep the maximum number of suprogram parameters within the limits of human understanding capacity.[10] The upper limit should be somewhere around seven variables, plus or minus two.

For the programmer using a subprogram, there is less to goof up when setting up and calling the subprogram if the parameter list is small. Also, if the list is small, the programmer initially writing the subprogram has an easier job.

Separate Inputs from Outputs

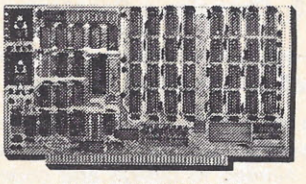
Separating which parameters are needed for input from parameters given as output greatly increases the subprogram's usability. Separating inputs from outputs clearly dis-

tinguishes "what is needed" from "what is produced." Any variable contributing both as an input and output parameter should be in the middle of the parameter list. Separation should also be done when defining parameters (via REAL, INTEGER, etc.). Inputs should be defined first, then parameters having both input and output characteristics should appear, and finally output parameters are defined last.

Document Uses of Static/Common Storage Requirements

When the time arises, due to practicalities or by the design solution itself, to pass data by means other than the explicit interface, document the occurrence. Two additional sections in the comment area can document this special communication. The first section describes the COMMON region used by the subprogram. The second comment section describes any variables which retain their values as a result of the previous time the subprogram was called. These will be termed STATIC variables. A good example of a situation requiring a static variable is a "random" number generator subprogram (Figure 12).

```
00100   C----------------------------------------
00200   C  NAME: RANDOM NUMER GENERATOR
00300   C  PURPOSE: TEST RANDOM NUMBER GENERATOR SUBPROGRAM
00400   C  DESCRIPTION:
00500   C       SET LIMIT OF 100 CYCLES
00600   C       GENERATE 5 "RANDOM" NUMBERS PER CYCLE
00700   C       PAUSE AFTER EACH CYCLE
00800   C  DEVICES:
00900   C       TTY - CONSOLE TELETYPEWRITER
01000   C       KEY - KEYBOARD
01100   C  NOTES: NONE
01200   C----------------------------------------
01300   C
01400           INTEGER KEY, TTY
01500           INTEGER LIMIT, CYCLE, COUNT, SCROLL
01600           REAL NUMBER
01700           DATA TTY/1/, KEY/1/
01800   C
01900   C
02000   C
02100           WRITE(TTY,10)
02200   10      FORMAT(1H ,'START OF PROGRAM:')
02300   C
02400           LIMIT = 100
02500           DO 400 CYCLE = 1 , LIMIT
02600              DO 200 COUNT = 1 , 5
02700                 CALL RANDOM(NUMBER)
02800                 WRITE(TTY,100) NUMBER
02900   100           FORMAT(1H ,'NUMBER=',F10.4)
03000   200        CONTINUE
03100              WRITE(TTY,250)
03200   250        FORMAT(1H )
03300              READ(KEY,300) SCROLL
03400   300        FORMAT(I1)
03500   400     CONTINUE
03600   C
03700           WRITE(TTY,9999)
03800   9999    FORMAT(1H ,'END OF PROGRAM:')
03900           STOP
04000           END
04100   C
04200   C----------------------------------------
04300   C  NAME: RANDOM
04400   C  PURPOSE: COMPUTE NEXT RANDOM NUMBER
04500   C  CALLING SEQUENCE:
04600   C       CALL RANDOM(NEXT-NUMBER)
04700   C  INPUT ARGUMENTS:
04800   C       NONE
04900   C  OUTPUT ARGUMENTS:
05000   C       NEXT-NUMBER      NEXT RANDOM NUMBER
05100   C  DESCRIPTION:
05200   C       CALCULATES THE NEXT "RANDOM" NUMBER BASED ON
05300   C       THE PREVIOUS NUMBER GENERATED.  THE NUMBER IS
05400   C       GENERATED USING A MULTIPLIER AND A SEED.
05500   C  STATIC STORAGE:
05600   C       MULT             FIXED VALUE MULTIPLIER.
05700   C       SEED             SEED TO DETERMINE NEXT
05800   C                        RANDOM NUMBER (CHANGES
05900   C                        AFTER EACH NUMBER IS
06000   C                        COMPUTED.
06100   C  NOTES: NONE
06200   C----------------------------------------
06300   C
06400           SUBROUTINE RANDOM(NUMBER)
06500           REAL NUMBER
```

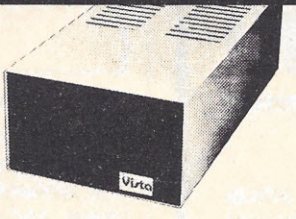
64K BYTE EXPANDABLE RAM

DYNAMIC RAM WITH ON BOARD TRANSPARENT REFRESH GUARANTEED TO OPERATE IN NORTHSTAR, CROMEMCO, VECTOR GRAPHICS, SOL, AND OTHER 8080 OR Z-80 BASED S100 SYSTEMS ★ 4MHZ Z-80 WITH NO WAIT STATES.
- ★ SELECTABLE AND DESELECTABLE IN 4K INCREMENTS ON 4K ADDRESS BOUNDARIES.
- ★ LOW POWER—8 WATTS MAXIMUM.
- ★ 200NSEC 4116 RAMS.
- ★ FULL DOCUMENTATION.
- ★ ASSEMBLED AND TESTED BOARDS ARE GUARANTEED FOR ONE YEAR AND PURCHASE PRICE IS FULLY REFUNDABLE IF BOARD IS RETURNED UNDAMAGED WITHIN 14 DAYS.

	ASSEMBLED / TESTED
64KRAM	$595.00
48K RAM	$529.00
32K RAM	$459.00
16K RAM	$389.00
WITHOUT RAM CHIPS	$319.00



S100 MAINFRAME AND CARD CAGE
- ★ W/ SOLID FRONT PANEL $239.00
- ★ W/ CUTOUTS FOR 2 MINI-FLOPPIES $239.00
- ★ 30 AMP POWER SUPPLY $119.00



VISTA V-200 MINI-FLOPPY SYSTEM
- ★ S100 DOUBLE DENSITY CONTROLLER
- ★ 204 KBYTE CAPACITY FLOPPY DISK DRIVE WITH CASE & POWER SUPPLY
- ★ MODIFIED CPM OPERATING SYSTEM WITH EXTENDED BASIC $695.00
- ★ EXTRA DRIVE, CASE & POWER SUPPLY $395.00

16K X 1 DYNAMIC RAM

THE MK4116-3 IS A 16,384 BIT HIGH SPEED NMOS, DYNAMIC RAM. THEY ARE EQUIVALENT TO THE MOSTEK, TEXAS INSTRUMENTS, OR MOTOROLA 4116-3.
- ★ 200 NSEC ACCESS TIME, 375 NSEC CYCLE TIME.
- ★ 16 PIN TTL COMPATIBLE.
- ★ BURNED IN AND FULLY TESTED.
- ★ PARTS REPLACEMENT GUARANTEED FOR ONE YEAR.

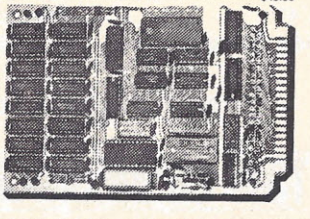
$8.50 EACH IN QUANTITIES OF 8

BETA COMPUTER DEVICES
1230 W. COLLINS AVE.
ORANGE, CA 92668
(714) 633-7280

KIM/SYM/AIM-65—32K EXPANDABLE RAM

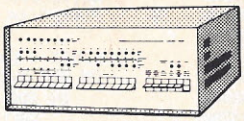
DYNAMIC RAM WITH ON BOARD TRANSPARANT REFRESH THAT IS COMPATIBLE WITH KIM/SYM/AIM-65 AND OTHER 6502 BASED MICROCOMPUTERS.
- ★ PLUG COMPATIBLE WITH KIM/SYM/AIM-65. MAY BE CONNECTED TO PET USING ADAPTOR CABLE. SS44-E BUS EDGE CONNECTOR.
- ★ USES +5V ONLY (SUPPLIED FROM HOST COMPUTER BUS). 4 WATTS MAXIMUM.
- ★ BOARD ADDRESSABLE IN 4K BYTE BLOCKS WHICH CAN BE INDEPENDENTLY PLACED ON 4K BYTE BOUNDARIES ANYWHERE IN A 64K BYTE ADDRESS SPACE.
- ★ BUS BUFFERED WITH 1 LS TTL LOAD.
- ★ 200NSEC 4116 RAMS.
- ★ FULL DOCUMENTATION
- ★ ASSEMBLED AND TESTED BOARDS ARE GUARANTEED FOR ONE YEAR, AND PURCHASE PRICE IS FULLY REFUNDABLE IF BOARD IS RETURNED UNDAMAGED WITHIN 14 DAYS.

	ASSEMBLED / TESTED
WITH 32K RAM	$419.00
WITH 16K RAM	$349.00
WITHOUT RAM CHIPS	$279.00
HARD TO GET PARTS ONLY (NO RAMS)	$109.00
BARE BOARD AND MANUAL	$49.00



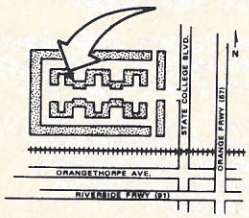
CALIF RESIDENTS PLEASE ADD 6% SALES TAX. MASTERCHARGE & VISA ACCEPTED. PLEASE ALLOW 14 DAYS FOR CHECKS TO CLEAR BANK. PHONE ORDERS WELCOME.

BITS N BYTES



MICROCOMPUTER SYSTEMS

- *Microcomputer Service*
- *Specialized Hardware Integration*
- *Hardware and Software Designs*
- *Software Development Systems to your Specifications*
- *Business System Hardware for Resale*
- *We Stock Industrial Microsystems, North Star, PerSci, Soroc, Tarbell, Vector Graphic*



BITS N BYTES
College Business Park
679 "D" S. State College Blvd.
Fullerton, Calif. 92631
(714) 879-8386
HOURS
11 A.M. – 6 P.M. M-F
Sat. – By Appt.

CIRCLE INQUIRY NO. 85

NOW IN STOCK!

COMPUCOLOR II
Model 3—8K User RAM $1595.00
Model 4—16K User RAM $1895.00
Model 5—32K User RAM $2195.00

COMPUCRUISE
Automobile Trip Computer
Model 44 $199.95
Model 41 (No Cruise Control) $159.95

CENTRONICS
Model 730-1 Parallel $995.00
Model 730-3 Serial $1045.00

COMPRINT
225 CPS Printers
Model 912-P $660.00
Model 912-S $699.00
Model 912-P/TRS 80 $690.00

VERBATIM
5.25" Soft Sector Disks 10 for $29.95
8" Triple Certified Soft Sector Disks 10 for $49.95

Write For Our Free Catalogue

T&W COMMUNICATIONS

10611 Harwin, Suite 406, Houston, Texas 77036
(713) 772-8301

Prices & availability subject to change without prior notice.

CIRCLE INQUIRY NO. 100

```
06600           INTEGER SEED, MULT, PRODCT
06700           DATA SEED/3729/, MULT/5167/
06800     C
06900     C
07000     C
07100           PRODCT = SEED * MULT
07200           SEED = PRODCT - (10000 * (PRODCT/10000))
07300           NUMBER = ABS(FLOAT(SEED) / 10000.0)
07400           RETURN
07500           END
```

Figure 12. Random Number Generator

Don't Pass in Constants

The programmer who uses a subprogram should never pass in a constant because there is *no* guarantee that the subprogram will not alter the value of any given argument. As shown by the program in Figure 13, a constant's value can be permanently changed by an inadvertent use of a subprogram or by an error in the subprogram itself. Some languages, like PL/I, guard the programmer from unexpected alteration of constants[11], however, most versions of FORTRAN will not protect the programmer from this disaster.

```
00100           INTEGER TTY
00200           REAL CONST
00300           TTY = 1
00400           CALL CHANGE(127.0)
00500           CONST = 127.0
00600           WRITE(TTY,10) CONST
00700     10    FORMAT(1H ,'CONSTANT=',F10.2)
00800           STOP
00900           END
01000     C
01100     C
01200           SUBROUTINE CHANGE(PARAM)
01300           REAL PARAM
01400           PARAM = PARAM + 1111.0
01500           RETURN
01600           END
```

Figure 13. Changing a Constant

Make Subprogram Do Only One Function

We should strive to write subprograms which perform only a single function. Having to do only one job certainly lessens the amount of effort needed to design and code a subprogram, and eliminates confusion and mistakes. Once coded, a subprogram performing a single function is easier to read and understand by others. It is also easier to test separately and integrate into a developing software system.

Keep Subprogram Size Small

Ensuring that a subprogram only does a single function is not always enough. If it requires 1000 lines of code to realize a particular function, that code should probably be broken into smaller pieces — each of which does a small part of the total job. The moral is that we should be aware of subprogram size — both in design and implementation — and break the subprogram into smaller pieces when it gets too unwieldy.

Use a Graphic Language to Design and Document Calling Structure

When creating a software system comprised of many subprograms, a graphic language should be used to represent the design solution and resulting calling structure. While many graphic languages are suitable for this, we will use HIPO to represent calling structures throughout this tutorial. In HIPO, a box represents a program or subprogram, and a line connecting two boxes represents an interface point. When a box is drawn for each subprogram, and a line is drawn for each interface, the resulting diagram provides a snapshot of the entire software system's structure (Figure 14).

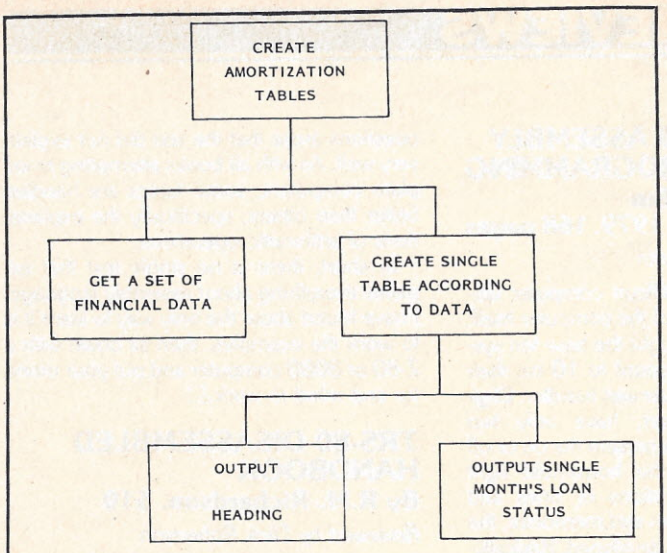


Figure 14a. HIPO Diagram of Amortization Program

```
00100   C-----------------------------------------
00200   C
00300           LOGICAL DONE
00400           INTEGER TTY,KEY,PTR
00500           REAL PRINC,RATE,PAYMNT
00600           DATA TTY/1/,KEY/1/,PTR/2/
00700   C
00800           CALL INPUT(PRINC,RATE,PAYMNT,DONE,TTY,KEY)
00900    100    IF (DONE) GO TO 200
01000              CALL CREATE(PRINC,RATE,PAYMNT,PTR)
01100              CALL INPUT(PRINC,RATE,PAYMNT,DONE,TTY,KEY)
01200              GO TO 100
01300    200    CONTINUE
01400   C
01500           STOP
01600           END
01700   C
01800   C-----------------------------------------
01900   C
02000           SUBROUTINE INPUT(PRINC,RATE,PAYMNT,DONE,TTY,KEY)
02100           LOGICAL DONE
02200           INTEGER TTY,KEY
02300           REAL PRINC,RATE,PAYMNT
02400   C
02500           WRITE(TTY,10)
02600    10     FORMAT(1H ,'PRINCIPLE:')
02700           READ(KEY,20) PRINC
02800    20     FORMAT(F10.2)
02900   C
03000           IF (.NOT.(PRINC .EQ. 0.0)) GO TO 30
03100              DONE = .TRUE.
03200              GO TO 60
03300    30     CONTINUE
03400              WRITE(TTY,40)
03500    40        FORMAT(1H ,'YEARLY RATE:')
03600              READ(KEY,20) RATE
03700              WRITE(TTY,50)
03800    50        FORMAT(1H ,'PAYMENT:')
03900              READ(KEY,20) PAYMNT
04000              DONE = .FALSE.
04100    60     CONTINUE
04200   C
04300           RETURN
04400           END
04500   C
04600   C-----------------------------------------
04700   C
04800           SUBROUTINE CREATE(PRINC,RATE,PAYMNT,PTR)
04900           INTEGER PTR
05000           REAL PRINC,RATE,PAYMNT
05100   C
05200           INTEGER MONTH
05300           REAL ACINT,MRATE,MOINT,MPRINC
05400   C
05500           MRATE = RATE / 12.0
05600           ACINT = 0.0
05700           MPRINC = PRINC
05800           MONTH = 1
05900   C
06000           CALL HEADNG(PRINC,MRATE,PAYMNT,PTR)
06100    10     IF (MPRINC .LE. 0.0) GO TO 20
06200              MOINT = MPRINC * MRATE
06300              ACINT = ACINT + MOINT
06400              MPRINC = MPRINC + MOINT - PAYMNT
06500              IF (MPRINC .LT. 0.0) MPRINC = 0.0
06600              CALL OUTPUT(MONTH,MPRINC,MOINT,ACINT,PTR)
06700              MONTH = MONTH + 1
06800              GO TO 10
06900    20     CONTINUE
07000   C
07100           WRITE(PTR,30)
07200    30     FORMAT(1H )
07300           RETURN
07400           END
07500   C
07600   C-----------------------------------------
07700   C
07800           SUBROUTINE HEADNG(PRINC,MRATE,PAYMNT,PTR)
07900           INTEGER PTR
08000           REAL PRINC,MRATE,PAYMNT
08100   C
08200           WRITE(PTR,10)
08300    10     FORMAT(1H ,'--------------------------------')
08400   C
08500           WRITE(PTR,20) PRINC
08600    20     FORMAT(1H ,'ORIGINAL PRINCIPLE:',F10.2)
08700           WRITE(PTR,21) MRATE
08800    21     FORMAT(1H ,'MONTHLY RATE:',F10.4)
08900           WRITE(PTR,22) PAYMNT
09000    22     FORMAT(1H ,'MONTHLY PAYMENT:',F10.2)
09100   C
09200           WRITE(PTR,10)
09300           WRITE(PTR,40)
09400    40     FORMAT(1H ,'MONTH      PRINC        INT       INT')
09500           WRITE(PTR,50)
09600    50     FORMAT(1H ,'                              PAID    ACCUM')
09700   C
09800           RETURN
09900           END
10000   C
10100   C-----------------------------------------
10200   C
10300           SUBROUTINE OUTPUT(MONTH,MPRINC,MOINT,ACINT,PTR)
10400           INTEGER MONTH,PTR
10500           REAL MPRINC,MOINT,ACINT
10600   C
10700           WRITE(PTR,10) MONTH,MPRINC,MOINT,ACINT
10800    10     FORMAT(1H ,I5,3F10.2)
10900   C
11000           RETURN
11100           END
```

Figure 14b. Amortization Program

NEXT MONTH

This chapter brings to a close our discussion of FORTRAN fundamentals. In future articles, we will apply the FORTRAN language to a host of practical everyday problems. Next in this series will be specific application of FORTRAN with data structure concepts to solve some basic sorting and searching problems.□

REFERENCES

1. Programming Languages: History and Fundamentals, J. Sammet, Prentice-Hall, 1969, pp. 145-146.
2. The Elements of Programming Style, B. Kernighan and P. Plauger, McGraw-Hill, 1974, p. 42.
3. On the Criteria to be Used in Decomposing Systems into Modules, D. Parnas, CACM, December 1972, pp. 131-136.
4. Structured Design, L. Constantine and E. Yourdon, Yourdon Press, 1975, pp. 500-501.
5. Top-down Structured Programming Techniques, C. McGowan and J. Kelly, Van Nostrand Reinhold Co., 1975, p. 19.
6. The Elements of Programming Style, B. Kernighan and P. Plauger, McGraw-Hill, 1974, n.p.
7. FORTRAN With Style,H. Ledgard and L. Chmura, Hayden Book Co., 1978, n.p.
8. Structured Design, L. Constantine and E. Yourdon, Yourdon Press, 1975, n.p.
9. Ibid., pp. 116-142.
10. The Magic Number Seven Plus or Minus Two, G. Miller, Psychology Review, 1956, pp. 81-97.
11. Compiler Construction for Digital Computers, D. Gries, John Wiley and Sons, 1971, pp. 189-190.

The Fantastic Intertec

SUPERBRAIN™

Complete computer, totally self-contained in a single box. 32K, 48K, or 64K versions. Uses two Z-80 CPU's. Commercial-type terminal w/12" monitor. Dual double-density minifloppies with 320 kilobytes of storage capacity. I/O port included. Expandable with an external S-100 bus interface. Comes with CP/M operating system. Extensive software support.

All this . . . and sells for UNDER $3,000!
Call for price and delivery



INTERTUBE II by Intertec
List $995 — **OUR PRICE $799**

12" display, 24x80 format, 18-key numeric keypad, 128 upper/lower case ASCII char. Reverse video, blinking. Complete cursor addressing and control. Special user-defined control function keys. Protected and unprotected fields. Line insert/delete and character insert/delete editing. Eleven special line drawing symbols.

OTHER VIDEO TERMINALS

PERKIN-ELMER BANTAM 550, $997. . . .	$799
with anti-glare screen, List $1027	$819
HAZELTINE 1400, List $850	$699
1410, List $900	$749
1420	$849
1500, List $1225	$989
1500 Kit, List $1125	$799
1510, List $1395	$1089
1520, List $1650	$1389
SOROC 120, List $995	$795
IQ140, List $1495	$1249
TELEVIDEO 912	$789

All printers below interface to TRS-80

CENTRONICS

779-1, no tractor, list $1245	$ 949
779-2 with tractor, List $1350	1049
702-2 with tractor VFU, List $2480	1995
703-2 with tractor VFU, List $2975	2395
730-1 parallel interface, List $995	895
T.I. 810 for TRS-80, List $1940	$1735
NEC Spinwriter for TRS-80, letter-quality, 55 cps	$2475

Disk Drives for TRS-80
Choice of leading brands
Only $369 each!

SHIPPING, HANDLING and INSURANCE: Add $10. All prices subject to change and all offers subject to withdrawal without notice. Above prices are for prepaid orders. Slightly higher prices prevail for other-than-prepaid orders.

BANKAMERICARD VISA write for free catalog master charge

MiniMicroMart
1618 James St., Syracuse NY 13203
(315) 422-6666 TWX 710-541-0431

126 INTERFACE AGE CIRCLE INQUIRY NO. 95

BOOK REVIEWS

Z-80 AND 8080 ASSEMBLY LANGUAGE PROGRAMMING

By Kathe Spracklen
Hayden Book Co. 1979. 168 pages

Reviewed by Jerry Martin

One of the more difficult computer languages to learn is that of the computer itself. Humans have been taught the base ten system since they could count to 10 on their fingers or toes, whichever was handier. Digital computers, however, have only two fingers or toes. These represent the on or off condition, similar to the household light switch. Since combinations of ones and zeros are difficult to learn and memorize, the assembly language was developed. Basically, this is nothing more than calling combinations of binary digits by a unique name.

Even with this simplification, some of the concepts of the operation of a computer are not easy to fathom. To compound the problem, few books have been available that discussed assembly language in a manner that could be understood without a lot of background in computers.

Kathe Spracklen states in the preface that introductory materials dealing with software emphasis has been very scarce. This book is an attempt to fill that gap. I should first warn those who would like to jump into assembly language that a firm understanding of a higher level language such as BASIC is necessary. I say this because Spracklen's book as well as almost all others refer to structures like arrays and then explain how they are implemented in assembly language.

Chapter One starts off the "Bits, Bytes, and Boolean Operators." These are the secret passwords into computer arithmetic and are not as mysterious or threatening as they may sound. The method of converting between our familiar decimal system and binary is clearly described. Addition and subtraction computer style are also presented with graphic examples.

As you proceed to the next chapter, a little more insight is gained on just how the computer operates internally. Registers, stack, program counter and load instructions are described. As the title of the book implies, both 8080 and Z-80 instructions are covered. Since the 8080 is a subset of the Z-80 instructions, confusion as to which you are dealing with is really not a problem.

As you proceed through the book you will learn how to instruct the computer to do arithmetic, branching, manipulate data, create data structures, deal with binary coded decimal, and efficiency. Five appendices are included starting at page 103. They are:

(A) 8080/Z-80 Conversion Chart and Instruction Summary;
(B) ASCII Code Summary;
(C) 8080 Disassembler;
(D) Z-80 Extension Disassembler; and
(E) Answers to Exercises.

The ultimate question is does this book teach you assembly language in 102 pages? To the extent that any book can teach anything, this one can. But some questions arose that the text did not explain very well. As with all books attempting to explain computers, some topics are handled better than others, specifically the explanations of arithmetic operations.

In short, there is no single text that explains everything about assembly language. I have found about the only way to learn it is to work the examples, then sit down with a Z-80 or 8080 computer and put your monitor and mind to work.□

TRS-80 DISASSEMBLED HANDBOOK

By R.M. Richardson. $10

Reviewed by Dick Robertson

Here is a small 10-chapter handbook that will revolutionize assembly language programming for all microcomputers that utilize the widespread Microsoft TRS-80 Level II BASIC and its derivatives, including: Apple, Pet, KIM, Heathkit and many others. Its very unique concept begins with a number of short demonstration programs that derive for the user *all* BASIC functions' CALL addresses in ROM plus a number of chapters demonstrating their use in simple assembly language programs.

Any knowledgeable Level II BASIC programmer, with only a smattering of assembly language experience, will find this new approach to assembly language programming saving them 6 months to a year's study and practice on their way to mastering this heretofore difficult subject.

The handbook is written in a very readable informal style, and includes questions and answers for each chapter that allows each reader to self-program their progress and set their own pace. An All-Base Conversion program is included as a learning aid.

This is the first conversion program we have seen that includes both-way conversions for: decimal-binary-hex-split decimal-split hex, up to 65535 decimal. It alone is worth the handbook's price.□

TRS-80 DISK AND OTHER MYSTERIES

By H.C. Pennington
130 pages. $22.50

To the user who has a stack of vital disks that have become unreadable, for one reason or another, this book may well be worth its weight in gold, even at today's prices.

The book is much more than a simple "how-to" book. It gives the novice programmer a relatively painless insight into binary and hexadecimal notation, then proceeds to explain exactly how the disk is laid out, so the reader is never left to wonder how or why a particular zap is applied to correct such problems as unreadable directory; accidentally "killed" file; direct statement in file; physically damaged disk; bad parity errors; "lost" Electric Pencil™ files, and much more.

The book is well illustrated and all "fixes" are amply documented with directory printouts, graphs, charts and diagrams.□

Start Computing For Just $129.95 With An 8085-Based Professional Computer Kit—

Explorer/85

100% compatible with all 8080A and 8085 software & development tools!

No matter what your future computing plans may be, Level "A"—at $129.95—is your starting point.

Starting at just $129.95 for a Level "A" operating system, you can now build the exact computer you want. Explorer/85 can be your beginner's system, OEM controller, or IBM-formatted 8" disk small business system...yet you're never forced to spend a penny for a component or feature you don't want and you can expand in small, affordable steps!

Now, for just $129.95, you can own the first level of a fully expandable computer with professional capabilities—a computer which features the advanced Intel 8085 cpu, *thereby giving you immediate access to all software and development tools that exist for both the 8085 and its 8080A predecessor (they are 100% software compatible)*—a computer which features onboard S-100 bus expansion—plus instant conversion to mass storage disk memory with either 5-1/4" diskettes or standard IBM-formatted 8" disks.

For just $129.95 (plus the cost of a power supply, keyboard/terminal and RF modulator, if you don't have them already), Explorer/85 lets you begin *computing* on a significant level... *applying* the principles discussed in leading computer magazines...*developing* "state of the art" computer solutions for both the industrial and leisure environment.

Level "A" Specifications

Explorer/85's Level "A" system features the advanced Intel 8085 cpu, an 8355 ROM with 2k deluxe monitor/operating system, and an 8155 ROM-I/O—all on a single motherboard with room for RAM/ROM/PROM/EPROM and S-100 expansion, *plus* generous prototyping space.

(Level "A" makes a perfect OEM controller for industrial applications and is available in a special Hex Version which can be programmed using the Netronics Hex Keypad/Display.)



Level "A" at $129.95 is a complete operating system, perfect for beginners, hobbiests, or industrial controller use.

PC Board: glass epoxy, plated through holes with solder mask • **I/O:** provisions for 25-pin (DB25) connector for terminal serial I/O, which can also support a paper tape reader ...provision for 24-pin DIP socket for hex keyboard/display...cassette tape recorder input...cassette tape recorder output...cassette tape control output...speaker output... LED output indicator on SOD (serial output) line...printer interface (less drivers)...total of four 8-bit plus one 6-bit I/O ports • **Crystal Frequency:** 6.144 MHz • **Control Switches:** reset and user (RST 7.5) interrupt...additional provisions for RST 5.5, 6.5 and TRAP interrupts onboard • **Counter/Timer:** programmable, 14-bit binary • **System RAM:** 256 bytes located at F8ØØ, ideal for smaller systems and for use as an isolated stack area in expanded systems...RAM expandable to 64k via S-100 bus or 4K on motherboard.

System Monitor (Terminal Version): 2k bytes of deluxe system monitor ROM located at FØØØ leaving ØØØØ free for user RAM/ROM. Features include tape load with labeling ...tape dump with labeling...examine/change contents of memory ...insert data...warm start...examine and change all registers...single step with register display at each break point, a debugging/training feature...go to execution address... move blocks of memory from one location to another...fill blocks of memory with a constant...display blocks of memory ...automatic baud rate selection...variable display line length control (1-255 characters/line)...channelized I/O monitor routine with 8-bit parallel output for high speed printer... serial console in and console out channel so that monitor can communicate with I/O ports.

System Monitor (Hex Version): Tape load with labeling... tape dump with labeling...examine/change contents of memory...insert data...warm start...examine and change all registers...single step with register display at each break point ...go to execution address. Level "A" in the *Hex Version* makes a perfect controller for industrial applications and can be programmed using the Netronics Hex Keypad/Display.



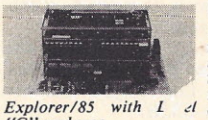
Hex Keypad/Display.

Hex Keypad/Display Specifications

Calculator type keypad with 24 system defined and 16 user defined keys. 6 digit calculator type display which displays full address plus data as well as register and status information.

Level "B" Specifications

Level"B" provides the S-100 signals plus buffers/drivers to support up to six S-100 bus boards and includes: address decoding for onboard 4k RAM expansion select-able in 4k blocks...address decoding for onboard 8k EPROM expansion selectable in 8k blocks...address and data bus drivers for onboard expansion...wait state generator (jumper selectable), to allow the use of slower memories...two separate 5 volt regulators.



Explorer/85 with Level "C" card cage.

Level "C" Specifications

Level "C" expands Explorer's motherboard with a card cage, allowing you to plug up to six S-100 cards directly into the motherboard. Both cage and cards are neatly contained inside Explorer's deluxe steel cabinet. Level "C" includes a sheet metal superstructure, a 5-card gold plated S-100 extension PC board which plugs into the motherboard. Just add required number of S-100 connectors

Level "D" Specifications

Level "D" provides 4k or RAM, power supply regulation, filtering decoupling components and sockets to expand your Explorer/85 memory to 4k (plus the original 256 bytes located in the 8155A). The static RAM can be located anywhere from ØØØØ to EFFF in 4k blocks.

Level "E" Specifications

Level "E" adds sockets for 8k of EPROM to use the popular Intel 2716 or the TI 2516. It includes all sockets, power supply regulator, heat sink, filtering and decoupling components. Sockets may also be used for soon to be available RAM IC's (allowing for up to 12k of onboard RAM).

Order A Coordinated Explorer/85 Applications Pak!

Experimenter's Pak (SAVE $12.50)—Buy Level "A" and Hex Keypad/Display for **$199.90** and get FREE Intel 8085 user's manual plus FREE postage & handling!

Student Pak (SAVE $24.45)—Buy Level "A," ASCII Keyboard/Computer Terminal, and Power Supply for **$319.85** and get FREE RF Modulator plus FREE Intel 8085 user's manual plus FREE postage & handling!

Engineering Pak (SAVE $41.00)—Buy Levels "A," "B," "C," "D," and "E" with Power Supply, ASCII Keyboard/Computer Terminal, and six S-100 Bus Connectors for **$514.75** and get 10 FREE computer grade cassette tapes plus FREE 8085 user's manual plus FREE postage & handling!

Business Pak (SAVE $89.95)—Buy Explorer/85 Levels "A," "B," and "C" (with cabinet), Power Supply, ASCII Keyboard/Computer Terminal (with cabinet), 16k RAM, 12" Video Monitor, North Star 5-1/4" Disk Drive (includes North Star BASIC) with power supply and cabinet, all for just **$1599.40** and get 10 FREE 5-1/4" minidiskettes ($49.95 value) plus FREE 8085 user's manual plus FREE postage & handling!

Continental U.S.A. Credit Card Buyers Outside Connecticut

CALL TOLL FREE 800-243-7428

To Order From Connecticut Or For Technical Assistance, Etc. **Call (203) 354-9375**

Netronics R&D Ltd., Dept. RE 10
333 Litchfield Road, New Milford, CT 06776

Please send the items checked below—

☐ **Explorer/85 Level "A" Kit** (ASCII Version), **$129.95** plus $3 p&h.

☐ **Explorer/85 Level "A" Kit** (Hex Version), **$129.95** plus $3 p&h.

☐ **8k Microsoft BASIC** on cassette tape, **$64.95** postpaid.

☐ **8k Microsoft BASIC in ROM Kit** (requires Levels "B," "D," and "E"), **$99.95** plus $2 p&h.

☐ **Level "B" (S-100) Kit, $49.95** plus $2 p&h.

☐ **Level "C" (S-100 6-card expander) Kit, $39.95** plus $2 p&h.

☐ **Level "D" (4k RAM) Kit, $69.95** plus $2 p&h.

☐ **Level "E" (EPROM/ROM) Kit, $5.95** plus 50¢ p&h.

☐ **Deluxe Steel Cabinet** for Explorer/85, **$49.95** plus $3 p&h.

☐ **ASCII Keyboard/Computer Terminal Kit** (features a full 128 character set, upper & lower case, full cursor control, 75 ohm video output convertible to baudot output, selectable baud rate, RS232-C or 20 ma. I/O, 32 or 64 character by 16 line formats, and can be used with either a CRT monitor or a TV set (if you have an RF modulator), **$149.95** plus $2.50 p&h.

☐ **Hex Keypad/Display Kit, $69.95** plus $2 p&h.

☐ **Deluxe Steel Cabinet** for ASCII Keyboard/Terminal, **$19.95** plus $2.50 p&h.

☐ **Power Supply Kit** (±8V @ 5 amps) in deluxe steel cabinet, **$39.95** plus $2 p&h.

☐ **Gold Plated S-100 Bus Connectors, $4.85** each, postpaid.

☐ **RF Modulator Kit** (allows you to use your TV set as a monitor), **$8.95** postpaid.

☐ **16k RAM Kit** (S-100 Board expands to 64k), **$199.95** plus $2 p&h.

☐ **32k RAM Kit, $329.95** plus $2 p&h.

☐ **48K RAM Kit, $459.95** plus $2 p&h.

☐ **64k RAM Kit, $589.95** plus $2 p&h.

☐ **16k RAM Expansion Kit** (to expand any of the above up to 64k), **$139.95** plus $2 p&h each.

☐ **Intel 8085 cpu User's Manual, $7.50** postpaid.

☐ **Special Computer Grade Cassette Tapes, $1.90** each or **3 for $5,** postpaid.

☐ **12" Video Monitor** (10 MHz bandwidth), **$139.95** plus $5 p&h.

☐ **North Star Double Density Floppy Disk Kit** (One Drive) for Explorer/85 (includes 3 drive S-100 controller, DOS, and extended BASIC with personalized disk operating system—just plug it in and you're up and running!), **$699.95** plus $5 p&h.

☐ **Power Supply Kit** for North Star Disk Drive, **$39.95** plus $2 p&h.

☐ **Deluxe Case** for North Star Disk Drive, **$39.95** plus $2 p&h.

☐ **Experimenter's Pak** (see above), **$199.90** postpaid.

☐ **Student Pak** (see above), **$319.85** postpaid.

☐ **Engineering Pak** (see above), **$514.75** postpaid.

☐ **Business Pak** (see above), **$1599.40** postpaid.

Total Enclosed $ _____
(Conn. res. add sales tax) By—
☐ Personal Check ☐ M.O./Cashier's Check ☐ Visa ☐ Master Charge
(Bank # _ _ _ _)

Acct. # _____

Signature _____ Exp. Date ____

Print Name _____

Address _____

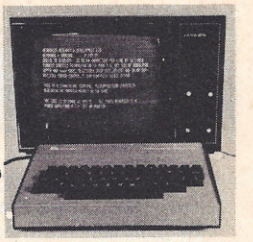
City _____

State _____ Zip _____

☐ **Send Me Information**

By Netronics

ASCII/BAUDOT, STAND ALONE



Computer Terminal

COMPLETE FOR ONLY... $149.95

The Netronics ASCII/BAUDOT Computer Terminal Kit is a microprocessor-controlled, stand alone keyboard/terminal requring no computer memory or software. It allows the use of either a 64 or 32 character by 16 line professional display format with selectable baud rate, RS232-C or 20 ma. output, full cursor control and 75 ohm composite video output.

The keyboard follows the standard typewriter configuration and generates the entire 128 character ASCII upper/lower case set with 96 printable characters. Features include onboard regulators, selectable parity, shift lock key, alpha lock jumper, a drive capability of one TTY load, and the ability to mate directly with almost any computer, including the new Explorer/85 and ELF products by Netronics.

The Computer Terminal requires no I/O mapping and includes 1k of memory, character generator, 2 key rollover, processor controlled cursor control, parallel ASCII/BAUDOT to serial conversion and serial to video processing—fully crystal controlled for superb accuracy. PC boards are the highest quality glass epoxy for the ultimate in reliability and long life.

VIDEO DISPLAY SPECIFICATIONS

The heart of the Netronics Computer Terminal is the microprocessor-controlled Netronics Video Display Board (VID) which allows the terminal to utilize either a parallel ASCII or BAUDOT signal source. The VID converts the parallel data to serial data which is then formatted to either RS232-C or 20 ma. current loop output, which can be connected to the serial I/O on your computer or other interface, i.e., Modem.

When connected to a computer, the computer must echo the character received. This data is received by the VID which processes the information, converting to data to video suitable to be displayed on a TV set (using an RF modulator) or on a video monitor. The VID generates the cursor, horizontal and vertical sync pulses and performs the housekeeping relative to which character and where it is to be displayed on the screen.

Video Output: *1.5 P/P into 75 ohm (EIA RS-170)* • **Baud Rate:** *110 and 300 ASCII* • **Outputs:** *RS232-C or 20 ma. current loop* • **ASCII Character Set:** *128 printable characters—*



BAUDOT Character Set: *A B C D E F G H I J K L M N O P Q R S T U V W X Y Z - ? : * 3 $ # () . , 9 0 1 4 ! 5 7 ; 2 / 6 8* • **Cursor Modes:** *Home, Backspace, Horizontal Tab, Line Feed, Vertical Tab, Carriage Return. Two special cursor sequences are provided for absolute and relative X-Y cursor addressing* • **Cursor Control:** *Erase, End of Line, Erase of Screen, Form Feed, Delete* • **Monitor Operation:** *50 or 60Hz (jumper selectable.*

Continental U.S.A. Credit Card Buyers Outside Connecticut

CALL TOLL FREE 800-243-7428

To Order From Connecticut Or For Technical Assistance, Etc. **Call (203) 354-9375**

Netronics R&D Ltd., Dept. PE-9
333 Litchfield Road, New Milford, CT 06776

Please send the items checked below—

☐ **Netronics Stand Alone ASCII Keyboard/Computer Terminal Kit, $149.95** plus $3.00 postage & handling.

☐ **Deluxe Steel Cabinet** for Netronics Keyboard/Terminal In Blue/Black Finish, **$19.95** plus $2.50 postage and handling.

☐ **Video Display Board Kit** alone (less keyboard), **$89.95** plus $3 postage & handling.

☐ **12" Video Monitor** (10 MHz bandwidth) fully assembled and tested, **$139.95** plus $5 postage and handling.

☐ **RF Modulator Kit** (to use your TV set for a monitor), **$8.95** postpaid.

☐ **5 amp Power Supply Kit** In Deluxe Steel Cabinet (±8VDC @ 5 amps, plus 6-8 VAC), **$39.95** plus $2 postage & handling.

Total Enclosed (Conn. res. add sales tax) $_____
By—
☐ Personal Check ☐ Cashiers Check/Money Order
☐ **Visa** ☐ **Master Charge** (Bank # _ _ _ _)

Acct. # _____

Signature _____ Exp. Date _____

Print Name _____

Address _____

City _____

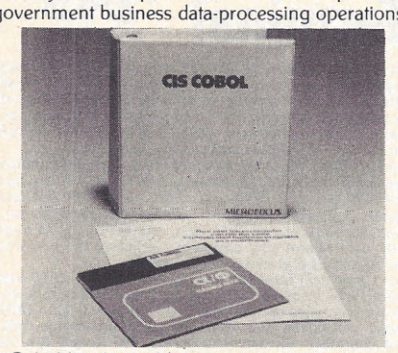
State _____ Zip _____

☐ **Send Me More Information**

NEW PRODUCTS

GSA Certifies CIS COBOL for Government Use

CIS Cobol from Micro Focus Inc., has become the first microcomputer Cobol system to win U.S. government certification. The certification opens the way for widespread use of microcomputers in government business data-processing operations.



Cobol has been the international industry standard language for commercial computer programming for twenty years. Although Federal Information Processing Standards Publication 21 makes it the language of choice in government business applications, and although there are a growing number of microcomputers being used in government, no certified version of the language existed until the General Services Administration (GSA) certified CIS Cobol.

For more information contact Micro Focus Inc., 1601 Civic Center Dr., Santa Clara, CA 95051, Chris Lloyd, (408) 984-6961.

CIRCLE INQUIRY NO. 121

Universal Software Language Brochure

A new eight-page color brochure describing PLMX, the first universal high-level language that communicates with all 8- or 16-bit microprocessors now available or yet to be developed, is being offered by Systems Consultants Inc.

The new software language not only generates code for any 8- or 16-bit microprocessor, but can also be adapted to interface with practically any operating system. Currently, the PLMX compiler can run under TEKDOS and CP/M operating systems. TEKDOS is the operating system for Tektronix' 8002A Universal Microprocessor Development System and CP/M is an operating system that can support just about any 8080-based system in use today, including many hobbyist and small industrial systems.

Copies of the PLMX brochure are available free from Systems Consultants Inc., Product Development Group, 4015 Hancock St., San Diego, CA 92110, (714) 292-PLMX.

CIRCLE INQUIRY NO. 122

Solid-State Temperature Controllers

The new E5H Series Temperature Controllers from Omron combine the low-maintenance benefits of solid-state circuitry with advanced performance features for easy maintenance, compact design and economical operation.



The ¼ DIN size units feature a space-saving housing that provides 2.4-inch behind-the-panel clearance.

For details contact Omron Electronics, Inc., 650 Woodfield, Schaumburg, IL 60195, (312) 843-7900, Robert Owens.

CIRCLE INQUIRY NO. 123

General Business Package

National Software Exchange, Inc. has announced version 2.0 of its General Business Package. This fourth generation product is a completely revised and enhanced version of its earlier package that includes General Ledger, Accounts Payable, Accounts Receivable, Payroll, and Inventory Control.

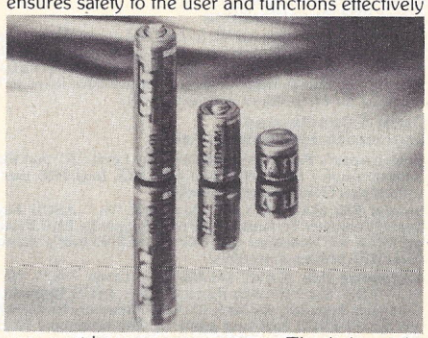
Now utilizing new facilities offered in CBASIC/2, the package is completely menu-driven and designed for ease of operation.

NSE sells only through authorized dealers who can supply source code on 8" or mini-diskettes. For details contact J.N. Lockley, 20 Ellerman Rd., Lake St. Louis, MO 63367.

CIRCLE INQUIRY NO. 124

Lithium/Copper Oxide Batteries

Available in three sizes (AA, ½AA and ¼AA), Saft's lithium/copper oxide cell has a proven ten-year shelf life as indicated by five-year desert storage data. The stability of this chemical system ensures safety to the user and functions effectively



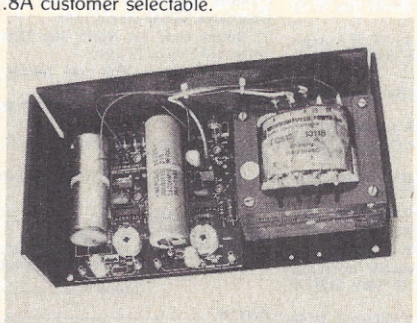
over a wide temperature range. The lightweight lithium system produces three to four times the energy of zinc carbon cells and twice as much as alkaline maganese cells.

For details contact Saft Corporation of America, 711 Industrial Blvd., Valdosta, GA 31601.

CIRCLE INQUIRY NO. 125

Dual Output Power Supply

The CC512 open-frame power supply is designed for digital or linear electronic systems requiring precision regulated d.c. power. DC outputs are 5V at 6A, and (9-15)V at 2.5A/or 5V at .8A customer selectable.



Specifications are 3mv peak to peak ripple, OVP built-in on 5V output, .05% line and load regulation, AC input 115/230VAC ±10%, and a transient response time of 30 useconds.

For details contact Microcomputer Power, Inc., 2272 Calle de Luna, Santa Clara, CA 95050.

CIRCLE INQUIRY NO. 126

ABC's Accounting Packages Now Available to DEC OEMs and Users

ABC Computers Inc. is making its business accounting system available as a separate software product. ABC's powerful and fully tested accounting software is now available to other OEMs or single users for use on DEC PDP 11 computers or ABC's "dekchester."

The packages are written in DIBOL and source codes are delivered with the packages. The programs are well documented and easy to follow. User manuals are available.

A "demo" license is available for evaluation of the package. For details contact ABC Computers Inc., Software Dept., P.O. Box 7529, Tahoe City, CA 95730.

CIRCLE INQUIRY NO. 127

Voice Input/Output Short Course

Voice input and output (VIO) devices range from educational toys to advanced applications in industrial automation and verbal computer data entry. Integrated Computer Systems has introduced a unique course on VIO.

The course, entitled "VIO — Voice Input/Output for Computer" emphasizes practical hardware and software implementations of speech synthesis and recognition algorithms. Topics covered in the course include fundamental speech characteristics, implementing voice output, recognizing words and continuous speech, applying available hardware and software systems and future trends. The course will be offered this spring in Los Angeles, Washington, D.C., London and other cities.

For details contact Integrated Computer Systems, Inc., 3304 Pico Blvd., Santa Monica, CA 90405, (213) 450-2060.

CIRCLE INQUIRY NO. 128

Nestar Multi-User System

The Clustershared™ personal computer system Cluster/One, Model A (for Apple), allows up to 64 Apple II computers to be tied together in a local network. Users may communicate with one another, share data, and access the same files — while the individual computer remains free to tackle problem solving tasks without being tied down by other computers in the system.

With up to 33 Mb of on-line storage in a single Model A, users may keep financial and operating records at their fingertips, may access training programs, or run statistical software. Memos and reports may be written, corrected, stored and 'mailed' to any other users on the Clustershared™ system.

For details contact Nestar Systems Inc., 430 Sherman Ave., Palo Alto, CA 94306.

CIRCLE INQUIRY NO. 129

Cartridge Disk for TRS-80 Model II

Cameo Data Systems announces a TRS-80 Model II Adapter for the Cameo DC-500 Cartridge Disk Controller.

Used with a Cameo controller, the new DC-504 Model II Adaptor allows attachment of up to four 2½-20 mbyte cartridge drives, thus giving the Radio Shack machine a large database capability.

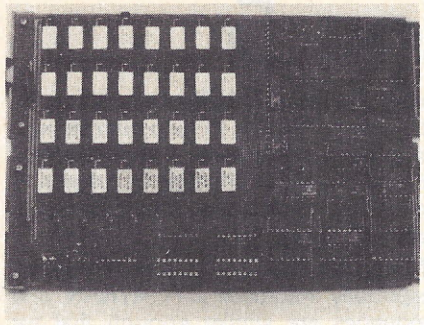
Removable cartridges facilitate multi-generation backup, needed to recover from program or operating errors, and can be used for archival storage as well.

The DC-500 Controller, DC-504 Adapter, and all cables are end-user priced at $1500. For information contact Cameo Data Systems Inc., 1626 Clementine St., Anaheim, CA 92802, (714) 535-1682.

CIRCLE INQUIRY NO. 130

64K Memory for H-11

Designed specifically for the Heathkit/Digital H-11 computer, LSI 11/2 and PDP 11/03 microcomputers is Chrislin Industries' CI-1103 memory module. The memory features an 8K x 16 dual width board using 200ns 4027 4K x 1 dynamic memory devices or a 32K x 16 dual



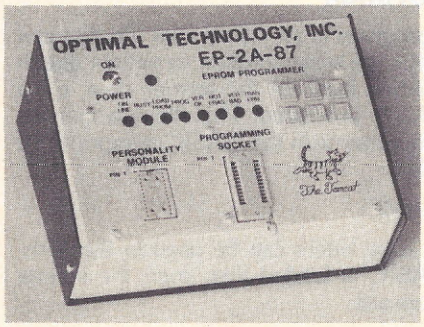
width board using 200ns 4116 16K dynamic memory devices. The unit simply plugs directly into the H-11, LSI-11, PDP 11/03 or LSI-11/2.

The CI-1103 is available with either on-board distributed refresh or external refresh control logic. Data access time is 300ns and cycle time is 525ns. For details contact Chrislin Industries, Inc., Computer Products Div., 31352 Via Colinas #102, Westlake Village, CA 91361, (213) 991-2254.

CIRCLE INQUIRY NO. 131

EPROM Programmer with RS-232 Interface

An EPROM Programmer, Model EP-2A-87, with RS-232 and 20 mA loop interfaces, includes a 2K or 4K buffer which can be loaded or read by another computer in the ON-LINE mode. Standard baud rates are 1200 and 100 which are dip switch selectable.



In the OFF-LINE mode, a keyboard enables the operator to program, verify, check if the EPROM is erased, and load the buffer from EPROM. Built-in self-test allows for checking the buffer and that the EPROM will tri-state.

For details contact Optimal Technology, Inc., Blue Wood 127, Earlysville, VA 22936.

CIRCLE INQUIRY NO. 132

6800 12K BASIC

Wintek's 12K BASIC is a full capability interpreter for the 6800 microprocessor. It supports three data types: numeric, string and boolean. Additional features including floating point, scientific function, and a random number generator make it ideal for engineering, scientific and data processing applications.

Direct memory read and write assembly language subroutines, and flexible input/output are oriented toward process control and data acquisition applications. User convenience features include interactive syntax checking, command completion and 21 error statements.

Available in cassette based and floppy disk based versions. For details contact Wintek Corp., 1801 South Street, Lafayette, IN 47904, (317) 742-8428.

CIRCLE INQUIRY NO. 133

Cross Assemblers

Lifeboat's XASM65 and XASM68 are two cross assemblers for the MOS technology 6502 and the Motorola MC6800 which can be used with the CP/M operating system. The assemblers will execute on 8080, 8085 or Z-80 microprocessors.

Each assembler supports the microprocessor manufacturer's mnemonics and will supply an optional symbol table and paginated listing.

For details contact Lifeboat Associates, 2248 Broadway, New York, NY 10024.

CIRCLE INQUIRY NO. 135

Multi-Function Numeric Keypad

The Multi-Function Numeric Keypad model KBAII is designed for the Apple II. The unit combines 10 numeric keys (w/double width 0) and 8 function keys: right/left cursor, minus, escape (ESC), slash (/), space, return (double-width) and period.

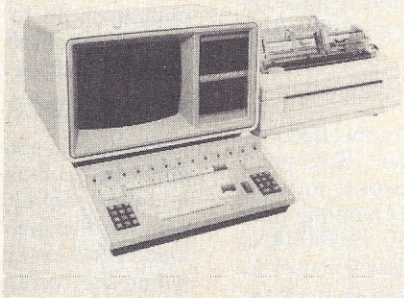
No modifications to the Apple II are required. The unit has easy plug-in installation and it may be used simultaneously with a computer keyboard.

The KABII list price is $199. For details contact California Micro Products, 795 W. Imperial Hwy., Brea, CA 92621, (714) 990-4014.

CIRCLE INQUIRY NO. 136

Word Processor/Communications Work Station

The 700/RTE mini-floppy word processor/communications work station is useful in applications that require two completely separate functions in one terminal system; stand-alone word processing and data processing with the capability to communicate with a central computer.



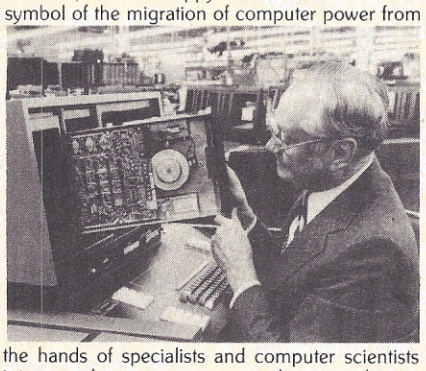
The 700/RTE is fully programmable and fully software controlled, and the screen attributes change automatically with the mode of operation.

For details contact Megadata Corp., 35 Orville Dr., Bohemia, NY 11716.

CIRCLE INQUIRY NO. 138

Data Processing Milestone

Shugart has delivered its one-half millionth SA800 series 8-inch floppy disk drive, marking a milestone for the data processing industry. The low-cost, versatile floppy disk drive is an evident symbol of the migration of computer power from



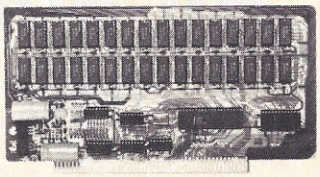
the hands of specialists and computer scientists into everyday commerce—even the private home.

The SA800 series of drives feature single or double density recording. All the electronics are contained on a single printed circuit card.

For details contact Shugart, 435 Oakmead Pkwy., Sunnyvale, CA 94086.

CIRCLE INQUIRY NO. 139

The days of complicated, unreliable, dynamic RAM are gone:



INTRODUCING

JAWS

the ultrabyte memory board

$199.95 *(complete kit with 16K memory)*

Netronics consistently offers innovative products at unbeatable prices. And here we go again — with JAWS, the ultrabyte 64K S100 memory board.

ONE CHIP DOES IT ALL

JAWS solves the problems of dynamic RAM with a state-of-the-art chip from Intel that does it *all*. Intel's single chip 64K dynamic RAM controller eliminates high-current logic parts . . . delay lines . . . massive heat sinks . . . unreliable trick circuits.

REMARKABLE FEATURES OF JAWS

Look what JAWS offers you: Hidden refresh . . . fast performance . . . low power consumption . . . latched data outputs . . . 200 NS 4116 RAMs . . . on-board crystal . . . 8K bank selectable . . . fully socketed . . . solder mask on both sides of board . . . designed for 8080, 8085, and Z80 bus signals . . . works in Explorer, Sol, Horizon, as well as all other well-designed S100 computers.

GIVE YOUR COMPUTER A BIG BYTE OF MEMORY POWER WITH JAWS — SAVE UP TO $90 ON INTRODUCTORY LIMITED-OFFER SPECIAL PRICES!

UNDECIDED? *TRY A WIRED 16K JAWS IN YOUR COMPUTER ON OUR 10-DAY MONEY- BACK OFFER (SPECIFY YOUR COMPUTER).*

CONTINENTAL U.S.A. CREDIT CARD BUYERS OUT SIDE CONNECTICUT CALL

CALL TOLL FREE 800-243-7428

From Connecticut Or For Assistance. (203) 354-9375

NETRONICS RESEARCH & DEVELOPMENT LTD. Dept. A

333 Litchfield Road, New Milford, CT 06776

Please send the items checked below:

- ☐ **JAWS 16K RAM kit,** No. 6416, $199.95.*
- ☐ **JAWS 16K RAM fully assembled, tested, burned in,** No. 6416W, $229.95.*
- ☐ **JAWS 32K RAM kit,** No. 6432, (reg. price $329.95), **SPECIAL PRICE $299.95.***
- ☐ **JAWS 32K RAM fully assembled, tested, burned in,** No. 6432W, (reg. price $369.95), **SPECIAL PRICE $339.95.***
- ☐ **JAWS 48K RAM kit,** No. 6448, (reg. price $459.95), **SPECIAL PRICE $399.95.***
- ☐ **JAWS 48K fully assembled, tested, burned in,** No. 6448W, (reg. price $509.95), **SPECIAL PRICE $449.95.***
- ☐ **JAWS 64K RAM kit,** No. 6464, (reg. price $589.95), **SPECIAL PRICE $499.95.***
- ☐ **JAWS 64K RAM fully assembled, tested, burned in,** No. 6464W, (reg. price $649.95), **SPECIAL PRICE $559.95.***
- ☐ **Expansion kit, JAWS 16K RAM module,** to expand any of the above in 16K blocks up to 64K, No. 16EXP, $129.95.*

*All prices plus $2 postage and handling. Connecticut residents add sales tax.

Total enclosed: $_____

☐ Personal Check ☐ Money order or Cashiers Check

☐ VISA ☐ MASTER CHARGE (Bank No. ____)

Acct. No. _____ Exp. Date _____

Signature _____

Print Name _____

Address _____

City _____

State _____ Zip _____

☐ Send me more information

CIRCLE INQUIRY NO. 97

An 8080/Z-80 Disk BASIC Microsoft's Version 5

Review by Alan R. Miller, Software Editor

ABSTRACT

Microsoft's version 5 BASIC contains features found on few microcomputer BASICs. These include:

1. long variable names (up to 40 characters)
2. typing of variables as REAL, INTEGER, DOUBLE PRECISION, and STRING
3. a WHILE . . .WEND construction
4. easy linkage to assembly language programs
5. automatic line numbering and renumbering
6. octal and hexadecimal representation
7. multiple statements per line and multiple lines per statement
8. line-number tracing during execution

The following review is not meant to be a document of all of the many features of Microsoft's BASIC. Nor is it meant to serve as an operator's manual, since the 120-page manual that is provided does this very well. Instead, this review will cover some of the more interesting features and show some examples of their use.

INTRODUCTION

All computer programs ultimately have to be converted into the binary machine language of the computer. But, since binary computer programs are difficult for people to write and just as difficult to understand, higher-level computer program languages have been developed. BASIC, FORTRAN, and Pascal are higher-level languages that allow the user to express his ideas in a style similar to that used in ordinary English and mathematics. E.g., the following BASIC statements direct a computer to do something useful:

```
PRINT HEIGHT
Z = X + Y
IF A = B THEN C = D ELSE R = 0
IF R < 9 GO TO 100
```

A collection of such expressions constitutes a SOURCE program written by the programmer.

A separate processor program is used to convert the user's source program into binary instructions that the computer can understand. FORTRAN processors usually compile the source program into an intermediate relocatable program. Then a loader program converts this into absolute binary code in memory. Typical BASIC processors, on the other hand, don't change the user's source program. Instead, they interpret each instruction as it is encountered.

CBASIC (reviewed in August/September, 1979) is different from the usual BASIC. It compiles the source program into an intermediate file which is then interpreted by a run-time monitor. Microsoft provides a compiling version of BASIC that will be reviewed in a forthcoming article. In this case, relocatable modules that are compatible with FORTRAN are produced. The tradeoff between interpreting and compiling is that interpreted programs are easier to develop, but compiled programs run much faster.

Interpretive BASIC is easy to use because the execution can be stopped at any time and the values of all the variables can then be printed out or changed. Then the program execution can be continued. Microsoft BASIC also allows the programmer to turn-on a trace feature to help with debugging. Then each line number is printed as the interpreter encounters it. But the interpretation process is inherently slower. Each statement in a loop that is executed 100 times, has to be re-interpreted 100 times.

Actually, FORTRAN and BASIC form a complementary pair of languages (especially Microsoft's FORTRAN and BASIC). The two languages are similar enough that for some tasks, a program can be written and debugged in BASIC. Then when it is running satisfactorily, it can be converted to FORTRAN to take advantage of the greater speed and use of local variables. For example, the random number generator given in the review of Microsoft FORTRAN (March, 1979) was developed by using Microsoft BASIC.

THE BASIC INTERPRETER

The name BASIC usually refers to the BASIC interpreter, but is sometimes used to refer to the user's source program. BASIC interpreters are available in all sizes from Tiny BASIC through Disk Extended BASIC. The actual interpreter, which remains in the computer memory during operation, may occupy from 3K to 24K bytes. The larger BASIC interpreters, such as Microsoft's CP/M Disk BASIC, contain many useful features such as a full text editor.

THE SOURCE PROGRAM

The user's source program is an ASCII disk file that can either be generated using the system editor, or written directly under control of BASIC itself. In the former case, a file with the filetype of BAS is produced. This file can then be loaded and executed together with the BASIC interpreter by typing the command line:

```
A>MBASIC FILENAME
```

Either upper or lower case letters can be used in the command. Alternately, BASIC is loaded first, then the source program is loaded afterward:

```
A>BASIC
• • •
Ok        <BASIC response>
LOAD "FILENAME"
```

In this case, the filename must be embedded in quotation marks and it must be spelled in upper-case letters. BASIC then loads the source file into memory. If the source file is in ASCII format, as it would be if it were written with the system editor, then it is converted to an encoded form.

An alternate method of generating a source program is to use the editor in BASIC itself. After BASIC is loaded, type the command:

```
AUTO
```

and the response will be:

```
10
```

The line numbers required by BASIC will now automatically appear at the beginning of each line. The default for the initial number and for the increment are both 10. These can be changed by adding arguments to the AUTO command. Thus a command of:

```
AUTO 1000,100
```

Cromemco incorporated
Tomorrow's Computers Today



CROMEMCO is the company that offers total systems from one source. Rated number one in reliability, too. You can be sure that informed buyers choose only CROMEMCO. Floppy disk based systems using high-density technology and even Multi-Megabyte hard disk systems. CROMEMCO is the one that delivers the most powerful systems for the lowest cost. They offer software like word processing, data base management, or choose from a wide selection of their higher-level languages like 32K Structured BASIC, RATFOR, COBOL, FORTRAN, Z-80 Assembler - and even a Multi-User BASIC for applications needing up to seven terminals.

Choose from three different systems

- **Z2-H COMPUTER HARD DISK**
- **SYSTEM THREE COMPUTER 8" DUAL FLOPPY**
- **SYSTEM TWO COMPUTER 5" DUAL FLOPPY**

Take your choice of peripherals:

- **HIGH-SPEED MATRIX PRINTER**
- **LETTER-QUALITY PRINTER**
- **LOW-COST PRINTER**
- **DELUXE CRT - VERY INTELLIGENT!**

When selecting a computer system, make sure you get the best - get CROMEMCO. Whether your application is:

** Business ** Education ** Industry ** Sophisticated personal use

When you purchase a CROMEMCO System you become a member of the most select group of computer owners - CROMEMCO SYSTEM Owners.

The best place to purchase your CROMEMCO Computer System is from Sunshine Computer, Inc. Whether you are in Southern California or like to purchase your computer products by mail, we offer total system support, starting at the time you select your own CROMEMCO System from us. Our support just begins at purchase time and extends to complete warranty and maintenance plans by our factory-trained technicians. Become a satisfied CROMEMCO System owner of ours, like many businesses, major corporations, government offices, well-known universities and colleges, and individuals have done.

Sunshine Computer Inc.

20710 South Leapwood Ave. • Carson, California 90746 (213) 515-1736

will start the number at line number 1000 and increment the numbers by 100. The maximum allowable line number is 65529.

If the user enters the source program under control of BASIC, there will be some automatic editing. Lower-case letters will be converted to upper except when embedded in quotation marks. The expression:

```
10 Y = X + 0.5
```

will be changed to

```
10 Y = X +  .5
```

to increase execution speed at the expense of appearance. Also, numbers will be explicitly typed where possible. For example, the expression:

```
20 A = 1.0 E-5 + 1.0/3D
```

will be changed to:

```
20 A = .00001 + 1!/3#
```

The RENUM command can be used to renumber the source program when lines are too close.

SAVING THE SOURCE PROGRAM

If the source program is generated with the BASIC editor, or altered by it after loading, then a SAVE command must be given before exiting BASIC. This will save the latest version as a disk file of filetype BAS. The directory of files currently on the disk should first be inspected so that a new name can be selected.

```
FILES     <list disk files>
• • •<files are listed>
SAVE "FILENAME" or
SAVE "FILENAME",A
```

Source files are normally saved in a compressed binary format. This form requires less disk space, and the file can be loaded more quickly. Two disadvantages of this format are that the binary file cannot be listed at the system level, and it cannot be readily altered by the system editor. The alternate save command contains a final A as shown in the second example above. In this case, the file is saved in ASCII form, and can be easily edited by the system editor.

It is important that the disk directory be checked prior to giving the SAVE command. The reason is that the SAVE command will overwrite an existing file of the same name, without giving a warning. Thus the original file will be lost.

Another approach is to rename the original file prior to the save:

```
NAME "FILENAME.BAS" AS "FILENAME.BAK"
SAVE "FILENAME",A
```

The NAME command can be used to rename any disk file. But in this case, the file extensions must be given. Files can also be deleted from the directory to make room for new programs:

```
KILL "FILENAME.EXT"
```

Obviously, this command should be used carefully. If it is necessary to change diskettes, it is extremely important that a system warm start be performed prior to saving a file. If this is not done, the BASIC program will be lost. The command is simply:

```
RESET
```

SYSTEM COMMANDS

A Control-C is the standard interrupt command for BASIC. It is used to abort a program listing, or a program execution. The command is also used to terminate the AUTO line numbering operation. But a Control-C is also interpreted as an abort command by the CP/M system software. Microsoft has solved this potential conflict by isolating the keyboard input from the CP/M system.

Some of the CP/M system commands are passed on directly, or are interpreted identically by BASIC. For example, a Control-U cancels the current line during console input. A Control-S freezes the output on the console screen during scrolling. A tab is entered as a Control-I.

The DEL or RUB key is used to correct the previously typed character. But in this case BASIC embeds the entry in a pair of backslash characters. A Control-R will retype the current input line, presenting it in its corrected form.

When a keyboard error is corrected by use of the DEL key, CP/M will typically print the mistake a second time, right next to the original character. After several errors have been corrected in this fashion, the input line is unreadable. A Control-R can be typed at this point to view the corrected form.

Microsoft has added an alternate error-correction command to effectively eliminate this problem. Typing a Control-H, the ASCII backspace character, will delete the prior keyboard entry. But, in this case, the cursor backs up on the video screen, effectively removing the character. Control-O is an additional control character. It is used to toggle the video terminal on and off during output.

LIST OUTPUT

A Control-P typed while at the CP/M system level will direct all console output to the list device as well as to the console. This feature is not available, however, while in Microsoft BASIC. Nevertheless, there are several ways that output can be sent to the list device either directly, or indirectly. The direct method is straightforward. It utilizes the BASIC LPRINT and LLIST commands:

```
LLIST 30-100  <list lines 30 to 100>
LPRINT NAMES$ <list the string variable NAMES>
LPRINT USING B$; TEMP <list variable TEMP>
```

For these commands, however, the console does not duplicate the list output.

The indirect method can be used if the IOBYTE, located at address 3, has been incorporated into the user's CP/M system software. The IOBYTE can be used to map the four logical I/O devices (console, list, punch and reader) into 16 different physical devices. The user has to write the software to perform the selection of the desired peripherals.

With the IOBYTE incorporated into the user's system, a BASIC program can be written and debugged using the system console. The PRINT and PRINT USING commands are used for output. A hard copy of the program can then be obtained by giving the LLIST command (whether the IOBYTE feature has been incorporated or not).

The BASIC editor does not incorporate a global replace command. If it did, then all PRINT commands could be replaced by LPRINT commands. Now, when the program could then be rerun, the output would appear at the printer. By altering the IOBYTE value, a hard copy of the program output can be obtained without changing the source program. The IOBYTE is changed by using the BASIC POKE command. Typing:

```
POKE 3,1 : RUN
```

changes the IOBYTE to 1 then runs the program. Console output now appears at the printer. At the conclusion of the run, the command:

```
POKE 3,0
```

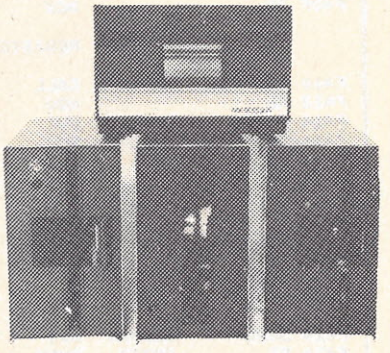
will reset the IOBYTE so that logical console output will appear at the console.

It is unfortunate that Microsoft does not allow the console and list devices to be used with the PRINT # statement. It

save on add-on products for..... TRS-80

TRS-80 COMPATABLE DISK DRIVES

The largest family of disk drives from the largest supplier, drives come complete with power supply and cabinet



TF-1 Pertec FD200, 40 track	$389
TF-3 Shugart SA400, same as tandy	$389
TF-5 MPI, B51 40 track	$379
TDF-1 Dual Head, 35 track	$499
MTI-40 Drive, 35 & 40 track	**$369**

220 versions available

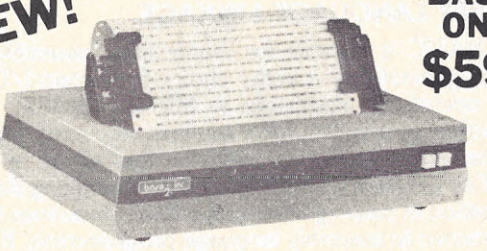
NOW! SUPER & MAX DISKS

SUPER: 77 tracks on one 5¼ diskette that's 197K of storage with New DOS+ $695

MAXI DISK II: 9.8 Megabyte on one disk using Winchester technology. Unit is sealed in unit for added protection $5349

LINE PRINTERS

NEW! **BASE 2 ONLY $599.**



The new base 2 printer, prints 60 lpm, 80 & 132 col. formats. It has tractor feed w/RS232, IEEE & centronic interface.

DP800 ANADEX 80 column, 112 cps	$949
LP779 Centronics 779	$1099
LP730 Centronics 730	$950
LP700 Centronics 700	$1395
LP701 Centronics 701	$1759
LP702 Centronics 702	$1895
Spin-1 Spinwriter	$2525

MOD II DISK DRIVES NOW AVAILABLE

ADD-ON DRIVE FOR ANY MICROCOMPUTER

DOES NOT INCLUDE POWER OR CABINET

Pertec FD200	$282
Pertec FD250 (dual head)	$349
Shugart SA400 (unused)	$286
Shugart SA800	$479
MPI B51	$279
MPI B52 (dual head)	$349

SOFTWARE

NEW DOS+ over 200 modifications, corrections and enhancements to **TRS DOS** Includes utilities.

35 Track NEWDOS+	$99
40 track NEWDOS+	$110
AJA Word Processor	$75
AJA Business Program	$250
Racet Infinite Basic	$49.95
Disk Drive Alignment Program	$109
Radex Data Base	$99.95
Electric Pencil	$150

NEW PRODUCTS

Small Systems Interface (RS232)	$49.95
Expansion Interface W32K	$499.95
AC Line Filter	$18.95
AC Isolater (6 connecters)	$47.95
Modem	$179.95
Verbatim 5¼ Diskettes	$3.39
16 Key Pad Kit (no soldering)	$68
16K Memory (Keyboard)	$89
16K Expansion (Interface)	$86

ALL PRICES CASH DISCOUNTED • FREIGHT FOB/FACTORY

MICROCOMPUTER TECHNOLOGY INCORPORATED
3304 West MacArthur
Santa Ana, CA 92704
(714) 979-9923

VISA
master charge THE INTERBANK CARD

Apparat, Inc.
7310 E. Princeton Ave.
Denver, CO 80222
(303) 758-7275

would make it much easier to divert output to the line printer. Xitan BASIC, for example, defines unit zero as the console and unit 2 as the list device. The user can initially define the variable LU to be zero. Output can then be sent to the console with the command:

PRINT #LU, <list of variables>

After the program is debugged, the output can be sent to the list device by redefining LU to be 2. In this method, only one variable has to be changed to switch the output from the console to the list.

ASSEMBLY LANGUAGE LINKAGE

While BASIC is suitable for many tasks, assembly language will generally be a better choice for certain I/O and memory operations. Provisions in Microsoft BASIC make linkage with assembly language routines very easy. Furthermore, the arrangement is virtually the same as with Microsoft's FORTRAN, COBOL, and the compiling BASIC.

The method of passing information into subroutines is one that is common to assembly language programming. If there is no argument then a simple CALL instruction is generated in BASIC. A RET instruction at the end of the assembly language program causes a return to BASIC. If the call from BASIC contains arguments:

110 CALL SORT(X,N)

then the protocol is more complicated. During the call sequence, the data is not actually passed to the subroutine. Instead, only memory pointers are passed. These pointers give the location of the actual data.

If the subroutine has no more than three arguments, the pointers are passed in the three double registers HL, DE, and BC, in that order. However, if there are more than three arguments, the register pair B,C points to a stack of pointers located in memory. A separate routine is provided in the FORTRAN and compiling BASIC packages that can make it easy to retrieve the stack of argument pointers in memory.

LISTING 1
Program to Print the Floating-Point Register

```
                ; PRINT THE FLOATING-POINT ACCUMULATOR
                ; FOR MICROSOFT BASIC
                ;
F400            ORG     0F400H
                ;
0010 =          CSTAT   EQU     10H     ;CONSOLE
0011 =          CDATA   EQU     CSTAT+1
                ;
                START:
F400 110300             LXI     D,3
F403 19                 DAD     D       ;POINT TO FAC
F404 CD10F4             CALL    BITS    ;EXPONENT
F407 2B                 DCX     H       ;MANTISSA
F408 CD10F4             CALL    BITS    ;HIGH BYTE
F40B 2B                 DCX     H
F40C CD10F4             CALL    BITS    ;MIDDLE BYTE
F40F 2B                 DCX     H       ;LOW BYTE
                ;
                ; PRINT REGISTER C IN ASCII BINARY
                ;
F410 CD24F4     BITS:   CALL    BLANK   ;SPACE
F413 4E                 MOV     C,M     ;GET BYTE
F414 0608               MVI     B,8     ;8 BITS
F416 79         BIT2:   MOV     A,C
F417 87                 ADD     A       ;SHIFT LEFT
F418 4F                 MOV     C,A     ;PUT BACK
F419 3E18               MVI     A,'0'/2 ;HALF OF 0
F41B 8F                 ADC     A       ;DOUBLE+CARRY
F41C CD26F4             CALL    OUTT    ;PRINT
F41F 05                 DCR     B
F420 C216F4             JNZ     BIT2    ;8 TIMES
F423 C9                 RET
                ;
F424 3E20       BLANK:  MVI     A,' '
F426 F5         OUTT:   PUSH    PSW
F427 DB10       OUTW:   IN      CSTAT   ;STATUS
F429 E602               ANI     2
F42B CA27F4             JZ      OUTW    ;NOT READY
F42E F1                 POP     PSW     ;GET BYTE
F42F D311               OUT     CDATA   ;SEND IT
F431 C9                 RET
                ;
F432                    END
```

A SAMPLE ASSEMBLY LANGUAGE LINKAGE

Microsoft BASIC, FORTRAN, and COBOL encode single-precision real numbers into four contiguous bytes. The lower-order three bytes contain the mantissa and sign of the number. The high-order byte contains the exponent. The assembly language program shown in Listing 1 can be used to learn more about how real numbers are encoded into a binary format.

The conversion program can be called by Microsoft BASIC, FORTRAN, or COBOL with one single-precision argument. Since there is only one argument, the H,L register pair contains the pointer to something called the floating-point accumulator, or FAC for short. This argument is actually the address of the lowest of the four bytes of the FAC, the least-significant byte of the mantissa.

Each time this assembly language program is called from BASIC with a real argument, the corresponding bit pattern of

LISTING 2
BASIC Program to Call Program in Listing 1

```
LIST
10 DEFINT I-N
20 FLOAT = &HF400
30 INPUT "A real number"; A
40 CALL FLOAT(A)
50 PRINT : GOTO 30
Ok
```

Mike's
announces
North Star Horizon Timesharing
with
Hard Disk Interface

In excess of 120 megabytes bulk storage capacity now possible. Several different hard disk units are available, interfaced to North Star DOS and BASIC.

Two to seven-user timesharing North Star Horizon integrated computer systems with: Dual density or quad capacity eight inch drives and/or hard disk units with your choice of a variety of printers.

A complete selection of innovative business application software is available for North Star* systems.

Foreign orders are our specialty.
Dealerships are available. Call or write for descriptive literature, hard disk availability and/or dealer information package.

Micro Mike's, Incorporated
905 South Buchanan * Amarillo, Texas 79101 * USA
Telephone: 806-372-3633

making technology uncomplicated — for People

CIRCLE INQUIRY NO. 93

Radio Hut Radio Hut Radio Hut Radio Hut

Radio Hut

201 LOCHWOOD MALL • DALLAS, TEXAS 75218
ORDER BY PHONE—214-324-5509

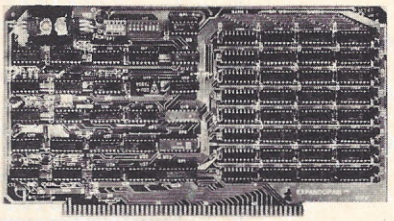
PLEASE WRITE FOR CATALOG
PRICES SUBJECT TO CHANGE
WITHOUT NOTICE

ORDERING INFORMATION & TERMS: Orders under $15.00 add 75¢ handling. No C.O.D. We accept Visa, MasterCharge, and American Express cards. Tex. Res. add 5% tax. Foreign orders (except Canada) add 20% P&H. 90 Day Money Back Guarantee on all items. Add 5% P&H, maximum $5.00.
ORDER BY PHONE — (214) 324-5509

SD EXPANDORAM

The EXPANDORAM is available in versions from 16K up to 64K, so for a minimum investment you can have a memory system that will grow with your needs. This is a dynamic memory with the invisible on-board refresh, and IT WORKS!

- Bank Selectable
- Phantom
- Power 8VDC, +16VDC, 5 Watts
- Lowest Cost Per Bit
- Uses Major Brand 16K RAMS
- PC Board is doubled solder masked and has silk-screened parts layout
- Extensive documentation clearly written



- Complete kit includes all Sockets for 64K
- Memory access time: 375ns, Cycle time: 500ns.
- No wait states required
- 16K boundaries and Protection, via Dip Switches
- Designed to work with Z-80, 8080, 8085 CPU's

EXPANDORAM 64K Kit (16K Ram)

16K	$239.95
32K	309.95
48K	379.95
64K	449.95

SD'S PROM 100 PROM Programmer Board

The PROM-100 Programmer is a development tool for S-100 Bus computer systems. The Zero Insertion Force Programming Socket extends above the card cage height for easy access to PROM devices. Software verifies PROM erasure, verifies program loading and provides for reading of object file from Disk or PROM and programming into PROM/EPROM. Features include: On-board generated 25vdc Programming pulse, TTL compatible, maximum programming time for 16,389 bits is 100 seconds. Programs: 2708, Intel 2758, 2716, 2732 and TI 2516. DIP Selectable EPROM type.

PROM-100 Board Kit $149.95

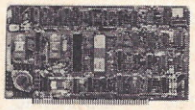
SD'S SBC-200 SINGLE BOARD COMPUTER Kit $289.95

S-100 Bus compatible and based on the powerful Z80 microprocessor, the SBC-200 meets the needs of a Z-80 CPU board with many additional features. Ideal for Industrial and control applications. All of the same features that have made the SBC-100 famous, PLUS 4MHz OPERATION. • S-100 Bus Compatible • Z80 Central Processing Unit • 1024 Bytes of Random Access Memory • 8K Bytes of PROM using 2716 • Serial Input/Output Port (with Asynchronous and Synchronous Operation) • Parallel Input and Output Ports • Four Channel Counter/Timer (Z80-CTC) • Software Programmable Baud Rate Generator • No-Front Panel Required for Operation • 4 MHz Operation.

SD'S VERSAFLOPPY II

• IBM 3740 Compatible Soft Sectored Format for Single Density Drives • Operates with Single and Dual Sided Drives, Single or Double Density Drives and 5" & 8" Drives — in any combination of four simultaneously • Drive Select and Side Select Circuitry • S-100 Bus Compatible • Vectored Interrupt Operation Optional • Phase Locked Loop Data Recovery Circuit • Operates with Z80 CPU's • Uses FD1791-1 Controller Chip • Ther Versafloppy II incorporates all the possible features of a flexible disk drive controller into one board. Capable of handling four drives simultaneously, combinations of any variety are possible, such as 5" single sided, 8" dual density dual sided, 5" dual density single sided. Most popular drives are controlled directly with the Versafloppy II. The operating system for the Versafloppy II is the extremely powerful SDOS available for SD Systems. Diagnostic and control software available to complete your disk system.

KIT $335.95



SD'S VDB-8024 VIDEO DISPLAY BOARD

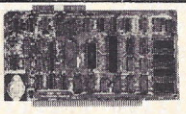
The VDB-8024 features its own on-board Z80 microprocessor. This gives the capability of using software (included in ROM) to control functions and enhancements without interference with the computer's CPU. Included in the special features: 80 characters by 24 lines display, keyboard power and interface, composite and separate video output, 2K on-board RAM, a total of 256 available characters, full cursor control, forward and reverse scrolling, underlining, field reverse, field protect enhancements, programmable characters.

KIT $329.95 A&T $389.95

SD'S "VERSAFLOPPY I" KIT

FEATURES: IBM 3740 soft sectored compatible, S-100 BNS Compatible for Z-80 or 8080. Controls up to 4 drives (single or double sided). Directly controls the following drives: Sugart SA400/450 Mini Floppy • Shugart SA800/850 Standard Floppy • PERSCI 70 and 277 • MFE 700/750 • CDC 9404/9406

$189.95



SD'S SBC-100 SINGLE BOARD COMPUTER

The SBC-100 provides a complete micro-computer on a single board! The Z80 microprocessor is used as the heart of the SBC-100. The SBC-100 meets all the requirements of a Z80 CPU board with the added features of I/O ports, counter/timer channels, on board RAM, provisions for PROM/ROM and a software programmable baud rate generator. S-100 Bus compatible, the SBC-100 features are: 8K bytes of available PROM, 1024 bytes on-board RAM, Serial I/O with both synchronous and asynchronous operation, Parallel I/O ports, Operational Vectored Interrupts, and Four Counter/Timer Channels. SD Monitor available for RS-232 and Video Terminals. Disk based system software also available.

SBC-100 KIT $249.95

TARBELL FLOPPY DISK INTERFACE

Compatible with Z80 & 8080. S-100 Bus. Uses CPM operating system. Plugs directly into your IMSAI or ALTAIR • Fastest transfer rate

KIT $190.00 Assembled & Tested $260.00

TARBELL CASSETTE INTERFACE

Plugs directly into your IMSAI or ALTAIR • Fastest transfer rate • Extremely reliable • Phase encoded • 4 extra status & control lines

KIT $99.95

CB2 Z-80 CPU Board Kit $186.00

Operates at 2 MHZ or 4 MHZ by DIP switch selection and includes two sockets for 2716 or 2732 EPROMs or TMS 4016 2K RAMs. Jumper options generate the new IEEE S-100 signals.

VB3 80 Character Video Board Kit $299.95

• 80 char. per line, up to 51 lines • Graphics up to 160 × 204 matrix • Up to 256 user defined symbols (optional EPROM) • Composite video

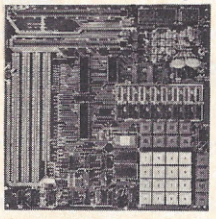
PB1 2708/2716 Programmer & 4K/8K EPROMBoard Kit $124.00

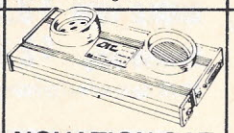
• S-100 bus • 2 separate programming sockets for 2708 or 2716 (5V) EPROMs • Programming voltage generated on board — no need for an external power supply • Software control of 2708/2716 programming selection • LED indicator for programming mode and an on-off switch for programming voltage • 4 sockets for 4K of 2708 or 8K of 2716 EPROMs • Unused EPROM sockets do not enable data bus drive so the board is never committed to the full 4K or 8K of memory • Jumper selectable wait states (0-4) for fast or slow EPROMS

Z80 STARTER KIT

Kit: $279.95 Assembled & Tested $349.95

SD System's Z80 Starter Kit enables the novice to build a complete microcomputer on a single board. Featuring the powerful Z80 microprocessor, the Z80 Starter Kit features • Keyboard and Display • Audio Interface • PROM Programmer • Expansion and Wire Wrap Area • On Board RAM • 4 Channel Counter/Timer • Z-BUG Monitor in PROM • I/O Ports.





NOVATION CAT ACOUSTIC COUPLER/MODEM $189.00

Let your computer communicate with other computers. Bell Systems 103 compatible 300 baud, answer or originate.

SYM-1

Reg. $269.00
NOW $219.00

- KIM-1 Compatible
- 4K ROM Monitor
- 1K Bytes 2114 RAM
- 65K Memory Expansion
- User EPROM 2716

IO4 2 Parallel/2 Serial I/O Board Kit $126.00

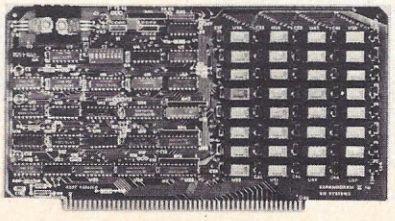
• S-100 bus • 2 serial I/O ports (2 in & 2 out) • Independent baud-rate selection from 55 to 9600 baud • Regulated +5V, +12V & −12V outputs provided on both serial headers • 2 latched parallel I/O ports (2 in & 2 out) • Independent DIP switches for setting address • Interrupt capability provided for on serial and parallel I/O ports • +8V @ 0.95A, +16V @ 0.6A, & −16V @ 80mA typical

VB1B VIDEO BOARD KIT $124.00

• S-100 bus • 64 or 32 characters per line (DIP switch selectable), 16 lines • Graphics 128×48 matrix • Upper case, lower case, Greek characters, symbols and numbers • 7×9 dot character matrix • Black-on-white or white-on-black • Timing 60Hz vertical rate, 16.2KHz horizontal rate, Crystal 12.44MHz • Parallel and composite video output (US TV signals), separate video, horizontal and vertical sync

SD'S EXPANDORAM II
The Randem Access Memory

- S-100 Bus Compatible
- Up to 4Mhz Operation
- Expandable Memory from 16K to 256K
- DIP Switch Selectable Boundaries
- Uses 16K (4116) or 64K (4164) Memory Devices
- Page Mode Operation Allows up to 8 Memory Boards on Bus
- Operates with Z80 CPU's
- Phantom Output Disable
- Invisible Refresh (Synchronized with Wait States)



SD Systems' ExpandoRAM II is a dynamic RAM board with capacities from 16K bytes (4116) to 256K bytes (4164). It operates on the industry S-100 Bus. The design allows 8 boards to operate from the same S-100 Bus. The ExpandoRAM II is compatible with most S-100 CPU's based on the Z80 microprocessor.

EXPANDORAM II KIT

16K	$295.95
32K	369.95
48K	444.95
64K	519.95

CIRCLE INQUIRY NO. 98

Radio Hut Radio Hut Radio Hut Radio Hut

LISTING 3

Binary Representation of the Floating-Point Accumulator

```
A real number? 0
 00000000 11000000 00000000 00000000
A real number? 1
 10000001 00000000 00000000 00000000
A real number? 2
 10000010 00000000 00000000 00000000
A real number? 3
 10000010 01000000 00000000 00000000
A real number? 4
 10000011 00000000 00000000 00000000
A real number? .1
 01111101 01001100 11001100 11001101
A real number? .2
 01111110 01001100 11001100 11001101
A real number? .3
 01111111 00011001 10011001 10011010
A real number? .5
 10000000 00000000 00000000 00000000
A real number? .499
 01111111 01111111 01111100 11101110
A real number? .25
 01111111 00000000 00000000 00000000
A real number? .2499
 01111110 01111111 11100101 11001010
A real number? -1
 10000001 10000000 00000000 00000000
A real number? -2
 10000010 10000000 00000000 00000000
A real number? 1E8
 10011011 00111110 10111100 00100000
A real number? 1E20
 11000011 00101101 01111000 11101101
A real number? 1E30
 11100100 01001001 11110010 11001101
A real number? 1E37
 11111011 01110000 10111101 11000111
A real number? 1E38
 11111111 00010110 01110110 10011100
A real number? 1E-38
 00000010 01011001 11000111 11011101
A real number? 1.3E38
 11111111 01000011 10011010 00110001
A real number? -1E38
 11111111 10010110 01110110 10011100
A real number? -1E-38
 00000010 11011001 11000111 11011101
A real number? 8
 10000100 00000000 00000000 00000000
```

the FAC will be displayed on the console. The 32 bits are arranged into four groups of 8-bit bytes. Listing 2 gives a BASIC program that will call the conversion routine each time the user enters a number. Listing 3 gives a sample session.

CONCLUSION

It is not the purpose of this review to serve as an operator's manual. The documentation provided by Microsoft is superb. Rather, some overall impressions and a few details of some of the more interesting features have been presented. The combination of the version 5 interpretive BASIC with the version 5 compiling BASIC produces a powerful package for general computing.

One of the few weaknesses of Microsoft BASIC (and all BASICs in general) is the lack of local variables in subroutines. This means that a routine written to sort the array X cannot be used directly to sort the array Y. Apparently, this problem will be overcome in a forthcoming version of Microsoft's compiling BASIC.□

DATA TERMINAL EQUIPMENT — FROM MICROMAIL



LA34 DECwriter IV

$999.00

- Upper/lower case, 9x7 dot matrix
- 10, 12, 13.2, 16.5 characters/inch
- 2, 3, 4, 6, 8 or 12 lines/inch
- 22"W x 7"H x 15½"D, 25 lbs.
- 110 or 300 baud, RS 232C serial ASCII
- Friction feed, paper width to 15"

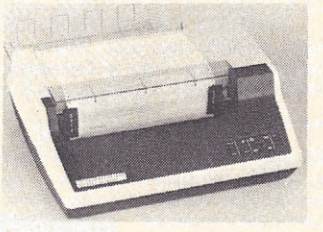
New from DIABLO

DIABLO 1640 $2,920.00
Receive-only $2,525.00
High-quality daisywheel printing at 45 cps.

DIABLO 1650 $3,070.00
Receive-only $2,675.00
Metal daisywheel printing at 40 cps.

T.I. 810 printer $1,599.00

- Includes upper/lower case
- 150 characters per second
- RS 232C serial interface
- Adjustable forms tractor



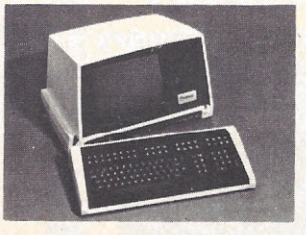
SOROC IQ 120 $740.00

- RS 232C, upper/lower case, full ASCII
- Numeric keypad, protected fields
- Cursor keys plus addressable cursor
- Auxiliary extension port



SOROC IQ 140 $1,130.00

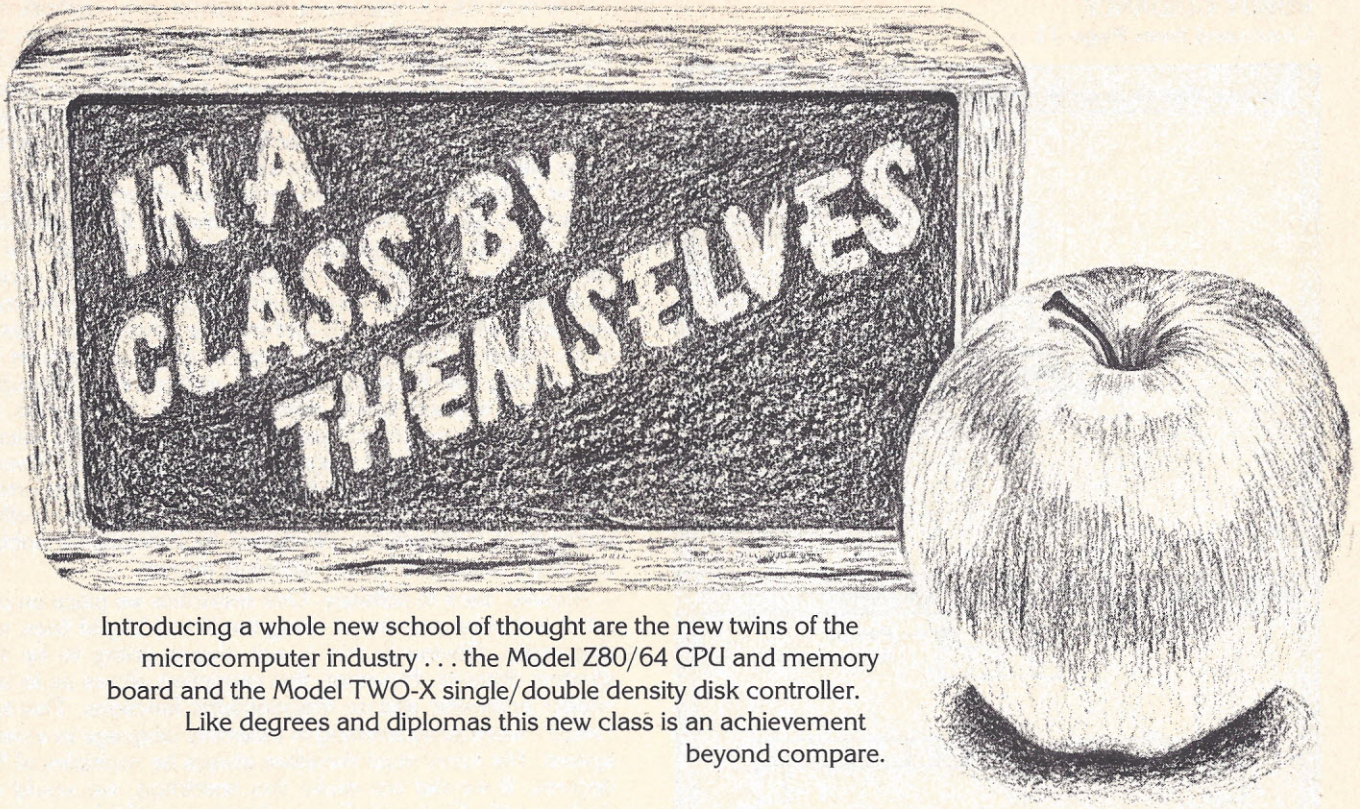
- RS 232C and 20mA current loop
- Extensive editing features
- 25th line terminal status display
- 16 function keys (32 with shift)



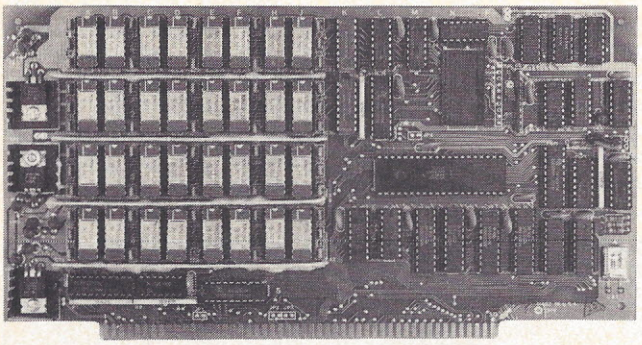
NEC Spinwriter
Call or write for prices

To Order: Send certified check (personal or company checks require two weeks to clear) including handling* and 6% sales tax if delivered within California.
*Handling: Less than $2,000, add 2%; over $2,000, add 1%. Everything shipped freight collect in factory cartons with manufacturer's warranty.

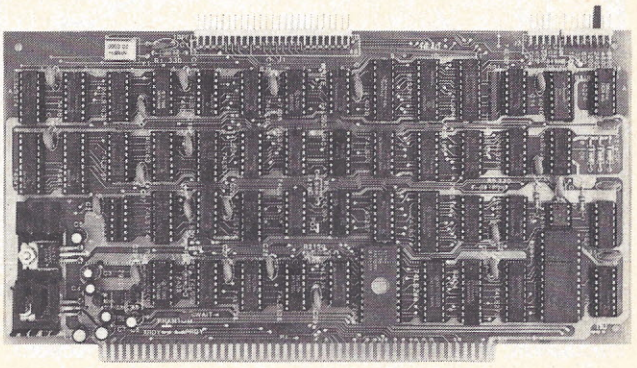
MICROMAIL™
MICROMAIL • BOX 3297 • SANTA ANA, CA 92703
(714) 731-4338



Introducing a whole new school of thought are the new twins of the microcomputer industry . . . the Model Z80/64 CPU and memory board and the Model TWO-X single/double density disk controller. Like degrees and diplomas this new class is an achievement beyond compare.



The Z80/64, a Z80 processor, runs at a full 4 Mz and is supplemented with 64K bytes of dynamic RAM with provisions for vectored interrupts and 2K EPROM, all on a quality S-100 bus board. The Z80/64 has low power requirements and full transparent refresh during unutilized processor time. With CPU and memory on the same board, timing problems and bus noise are eliminated. Who said you can't teach an old system new tricks?



The TWO-X, double/single density disk drive controller, handles four 8″ or three 5″ single or double headed disk drives, and operates single/double density automatically. Transfering files between densities, operating at 2 or 4 Mz, the TWO-X includes a RS232 Serial port. There's a hardware 2708 PROM on board and the software is all ready to go, configured to run without a front panel. It's a real apple polisher.

To send your system to the head of it's class, put these boards in our new "Cool" mainframe, the Model 2018 desk or rack mount cabinets . . . they're in a class by themselves.

So if you are ready to stand above the crowd, call the "Coolest" distributors in the country, CMC MARKETING, or see them at your local retail computer store.

CMC Marketing

10611 Harwin,Suite 406, Houston, Texas 77036 • (713) 995-4960 • TWX: 910-881-7155 TELEX: 762072

© 1979 CMC Marketing Corp.

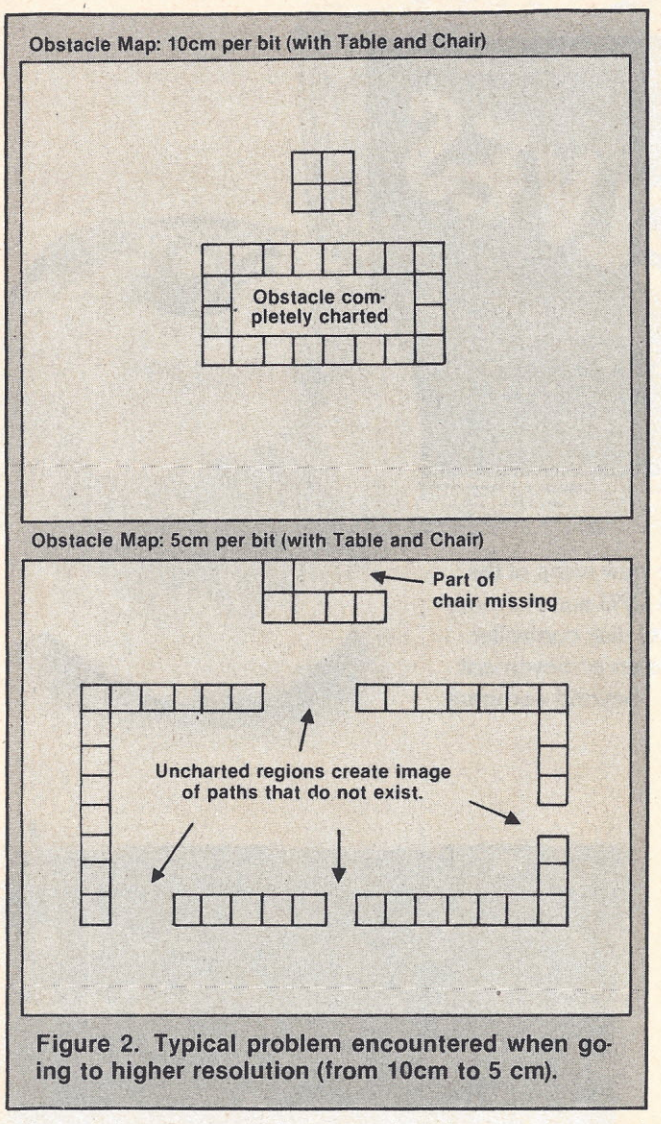


Figure 2. Typical problem encountered when going to higher resolution (from 10cm to 5 cm).

refers to determination of position based on mathematical calculation, without feedback from sensors for corrections. It is the easiest to implement, hardware-wise, but is less accurate than "Closed-loop," which is described later. The open-loop method is simplest to implement if stepping motors are used. With the formula:

$$L = d*PI*r/360$$

where: L is the distance travelled per step
d is the diameter of the wheel
PI is 3.1415926. . .
r is the number of degrees rotated per step

we can determine how far we have travelled for each step of the motor. For example, a 15 cm diameter wheel is rotated 30 degrees per step. This produces a 3.9269908 cm travel per step.

Immediately we see a problem. If this system is to be implemented in assembly language with INTEGER arithmetic only, we are in trouble with these calculations. How are you going to add successive increments of 3.9269908 cm to a counter in assembly language to deduce your position? You could round up to 4 cm per step, but notice that in an open-loop system all errors are cumulative. Taking a rounded-up 4 cm value, the distance travelled by 10 revolutions of the wheel comes out to 480 cm. Using the more precise value determined earlier, 10 revolutions only carried you 471.2 cm in the real world. If this is done all day long, Robot will be driving off balconies or pouring coffee into people's laps.

The best way to overcome this barrier is to calibrate the grid in terms of multiples of motor steps. This throws the memory requirement formula off a bit if you were looking for a value in feet or meters. You must instead think in motor step distances. It does not matter to the robot what units he is using, as long as *you* can interface with him on his terms every week or so as explained later.

The "Closed-loop" implies some type of sensor providing feedback to generate the position information. If you are using a DC motor (not a stepper) you have little way of telling how far the robot has travelled from start to stop of the motor, except by timing the run. You can mount a microswitch on the trailing swivel wheel (Figure 3) and make a "bump" or "dent" in the wheel surface every 45 degrees or so, creating an interrupt every time the switch is closed, and count the interrupts.

Another method which produces much less friction involves using a magnetic reed switch in place of the microswitch, mounting small magnets around the wheel. Mounting the sensor on the non-driven wheel will prevent errors due to slippage. It will be interesting to see what other methods of feedback are developed by enterprising hobbyists who choose not to use stepping motors.

Anyway, back to software. One restriction we place on our robot is that he only travel along the X and Y grid lines, not diagonally. Obviously this is not always going to be the shortest distance. However, this restriction allows us to use integer arithmetic and no trigonometric functions. This is a "must" when programming in assembly language in a small system. His turns must therefore always be multiples of 90 degrees. If we did not make this restriction, we would be dealing with Sine and Cosine functions, which are time-consuming and costly in terms of money.

So, now we have a way to determine position, and we need to find a way to also keep track of direction. Which way is the robot facing? The simplest way to track this is to use the same positional information supplied by the motor control described earlier. If it is an open-loop system, one can determine how far the vehicle has rotated by counting motor steps, since a turn is effected by rotating the motors in opposite directions. In a closed-loop system the feedback sensor provides the steps, and the software simply remembers that this is a turn, since the wheels are going in different directions.

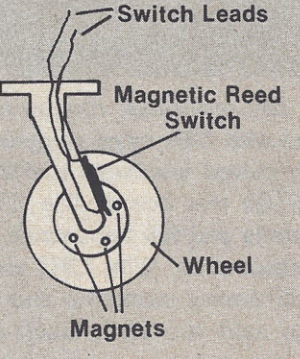


Figure 3. Using the trailing swivel wheel for motion feedback with a magnetic reed switch.

Now that he knows position and direction, you can say: "Proceed to coordinate 72,112 and turn facing North." He will calculate the distance between hither and yon, engage motors and leave dual track marks on the carpet as he speeds off and crashes into the wall. Important point: make sure you have microswitch bumpers on front, rear, and sides wired into fairly high priority interrupts that will quickly shut off the motors. Robot now recovers, shakes his head, and flips ON the bit corresponding to the location of the obstacle in memory. The situation is shown in Figure 4. The only part he can actually "see," however, is the shaded part; one bit.

Now, the first little bit of intelligence we give him is this:

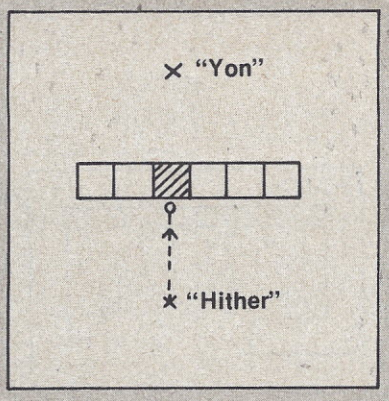


Figure 4. Initial collision with unknown obstacle between "hither" and "yon." (Only shaded portion is known after collision.)

before entering any coordinate, check that array location in the map and *do not* drive into it if it shows a '1.' So after that experience, if we drag Robot back to "hither" and tell him to go to "yon" again, he should drive up to just next to the wall and stop *without touching it.* There we may have programmed him to cry or give some other indication of cybernetic frustration.

To reach the destination, he will need some smarts to make him try alternate routes around the obstacle. Remember, so far he has no other information in "the world" than that there is one obstacle at that one location.

So, the next thing we provide him with is a routine for getting around what I call an "Open Obstacle" (Figure 5). Both ends are open and can be circumvented. Of course, he does not know yet what type of obstacle it is, but since this is the simplest, he must treat everything first as an Open Obstacle. (By the way, this system may be implemented in a phased, step-by-step manner by including the routines described, one by one, increasing the sophistication of the robot.)

Follow along on the diagram as you are reading. The theory here is really simple, but must be understood to go on to more complex obstacles.

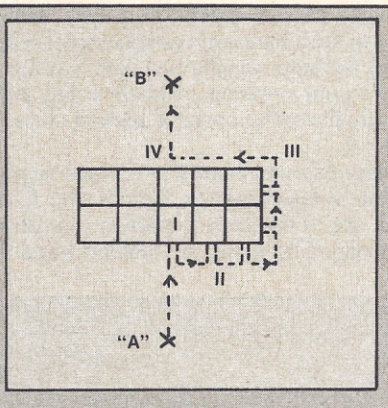


Figure 5. The "Open Obstacle."

Therefore, to avoid Open Obstacles, program the robot to back up just away from the wall, and pivot either left or right according to random choice (see path I). Then he must drive a short distance, stop, turn back in the original direction of travel and test to see if the obstacle is still there (II). If the obstacle is no longer there, he must drive straight ahead a short distance, stop, turn 90 degrees (in the opposite direction from the originally chosen course change) and "feel" if it is safe to turn back again. When it is (III), he may drive on until one of his coordinates matches that of the destination. Then he may pivot (IV) and head for the final target.

The idea of "testing" every few grid lines to see if the obstacle is still there is very central to this avoidance technique. This updates the array and soon the robot will not need to physically test a location; it will merely look at that array address in memory.

So our first obstacle avoidance problem was solved by a random movement to the right or left. Now we go on to the next logical step. As mentioned before, the robot does not *know* in advance that the wall may end three meters to the right. In the next case, let us assume that the obstacle actually continues all the way over to another wall (Figure 6). This I call a "Half-open Obstacle." There is only one simple way around it.

Applying the "Open Obstacle" pattern to get around the initial problem, the robot happens to choose to pivot to the right. (Had he chosen the left, it would have proven to be a simple Open Obstacle, as before.) Now, however, he goes and goes and cannot seem to find a way around it, finally bumping into the wall (at "W").

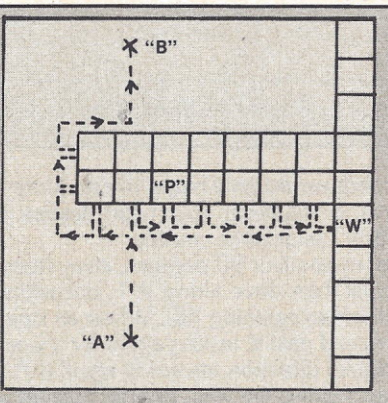


Figure 6. The "Half-open Obstacle."

Some limit could be built into this searching. Without it, he could attempt to go around the Great Wall of China the long way and never give up! A similar limit was apparently programmed into chickens, which will only go a finite distance out of their way to around a "wall" of glass to get food on the other side. They are so dedicated to that limit that they would rather die of starvation than to travel the extra inch it might take to peek around the corner of the glass obstacle.

Now, the robot has concluded that going to the right was not correct: he found another obstacle. He must therefore turn around and proceed back to the point of initial encounter with the wall at point "P," and take the other direction, in this case, to the left. He will now try the same thing on this side, and sure enough, it turns out to follow the Open Obstacle pattern from "P" on. Notice that the robot *remembered* what pattern he was in, until the obstacle was conquered. This is important for the more complicated maneuvers, and will be discussed at length later. The software must store the starting point and several crucial locations, such as "P," and the collision point with "W."

At this point we make an important observation: Once the obstacle has been mapped out, the software can "check out" any route before ever starting the motors. This way, the futile trip from P to W and back can be cut out of the final itinerary, making the robot quite a bit smarter. Obviously, without complete obstacle position information, the robot will make several futile trips while learning. However, he is no fool: He will not repeat the same mistake twice.

Armed with this amount of navigational skill, the robot can take on a fairly cluttered house. But there are some "fatal" traps. One of these I call the "Box" (Figure 7) because there is a blocking wall at *each* end of an open obstacle. Applying the standard routines just learned, he will first try, say, the left side (I), and, finding it blocked by wall W1, will head back to point P and try the right side (II). Finding it blocked, too, (at

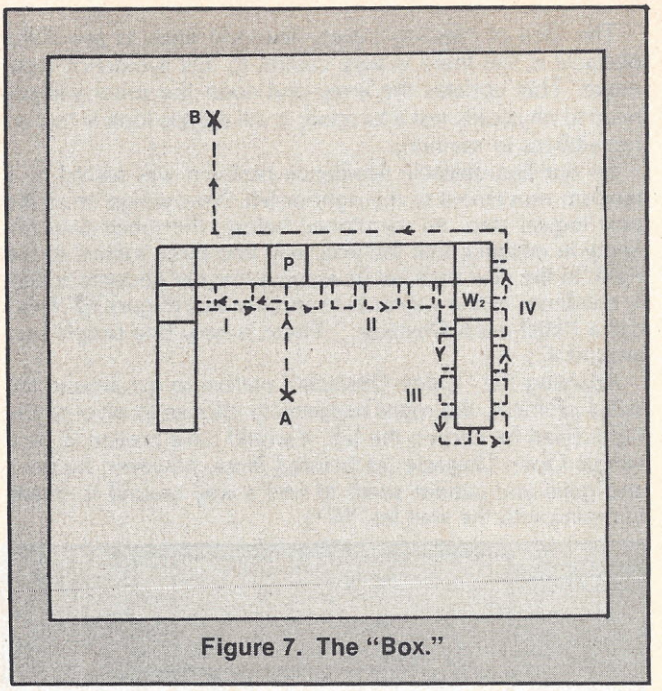


Figure 7. The "Box."

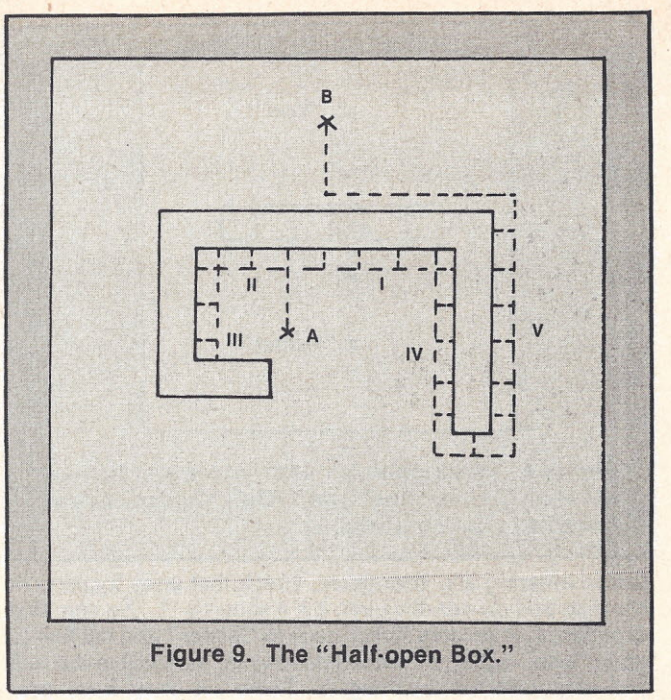


Figure 9. The "Half-open Box."

W2), our robot would now begin to cry, jump up and down, release flares, or whatever your heart desires. Here we must add our "Box" avoidance routine.

The robot must pivot 90 degrees, away from destination B totally. It must then drive along W1, occasionally turning to test the wall for an opening (III). When an opening is found, he must proceed past it, testing again for a way to cut back in the very original direction toward B again (IV). When found, he must travel along the wall, testing it for an opening (back to the left in our case) as indicated in path V. From the last corner it is smooth sailing to destination B.

Let us see how this applies to a real situation. Figure 8 shows a typical section of a house. Robot has been ordered to take Junior for a ride to his bedroom where Papa is waiting to put him in bed. (Only for strong robots!)

On the first leg of the trip, he encounters an Open Obstacle (I). Successfully past it, he finds a Box on leg II. Going through the doors, the bed at III forms a Half-open Obstacle, which finally leads him to Papa. By breaking down a total journey into individual obstacle patterns, each one "falls" on the way to the destination.

There are two more complicated patterns which should be included for maximum performance. The first of these actually already exists in the example above. It is called the "Half-open Box" because only one of the two sides is an escape route. The other side is closed by a wall (or other obstacle). Note on the diagram of Figure 8 that if the robot had chosen to creep along the wall W while attacking the Box at II, he would not have made it. So we have another rule: If in looking for the exit from a Box, the robot bumps into another obstacle, go back and try the other side (Figure 9).

The next strategy is a backup strategy and is essential for working mazes. This one should be built-in for unsolvable "problem obstacles." Simply stated, when all else fails, hug the nearest wall and follow every nook and cranny until you are past the obstacle.

I stated before that the software should remember the pattern it is currently executing. But for how long? At what point does one assume that that obstacle has been licked? Well, in examining previous patterns, we find that: *A pattern has been completed when the point of initial contact with the barrier has been passed, and one of the two coordinates of the destination has been reached.*

In some cases, the robot may be faced with an obstacle which cannot be circumvented. You may want to program the computer to terminate such unsuccessful searches for exits; maybe cry for help, maybe just *halt*. On the other hand, you may want your creation to keep trying until someone stops him manually or the obstacle pattern changes, allowing him to bypass it.

I have mentioned something about a type of pre-flight planning to be done by the computer. Here is why. Given the world in Figure 10, the robot may depart in one of two different directions initially. The paths are labeled I and II. Following

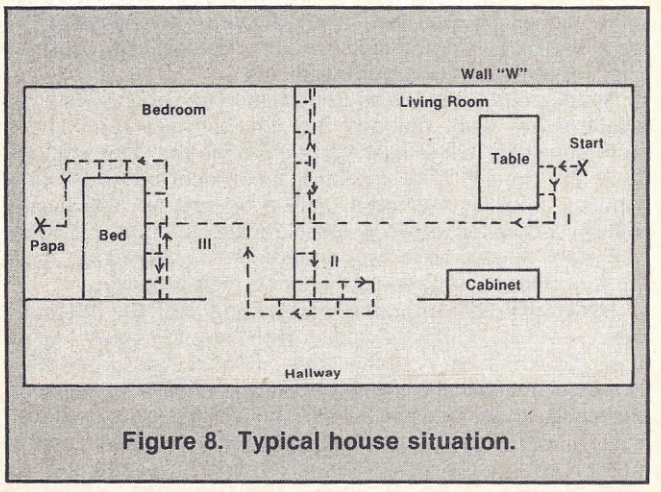


Figure 8. Typical house situation.

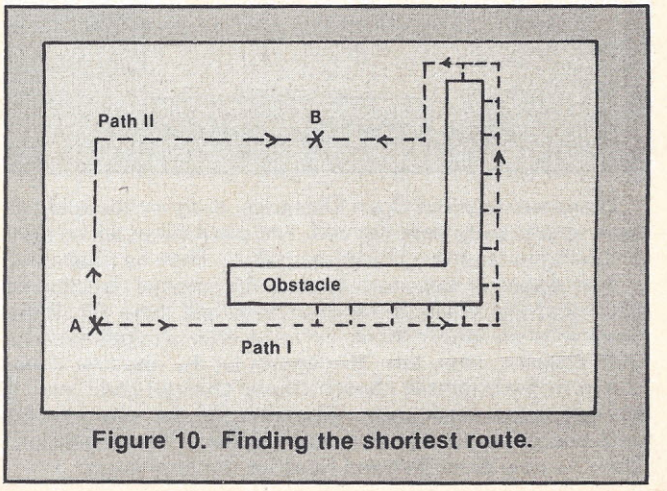


Figure 10. Finding the shortest route.

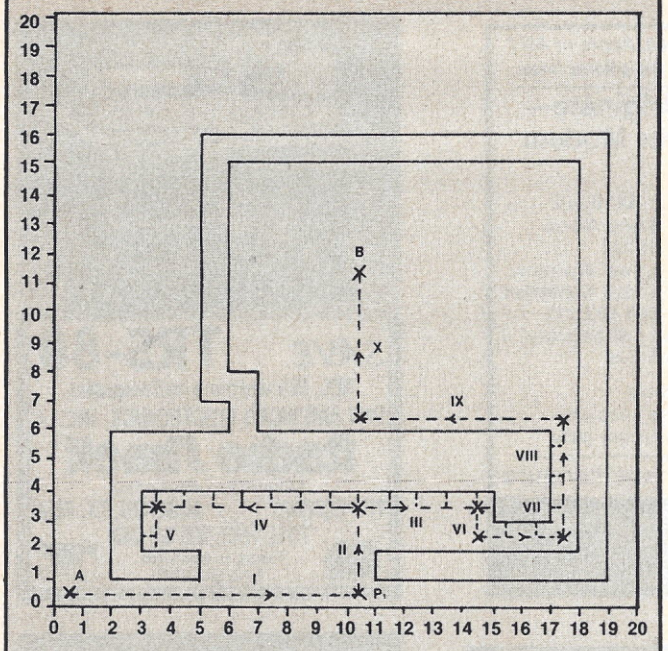


Figure 11. Path Optimization (See Table 1).

path I is obviously shorter than path II, but this must be determined by stepping through each of the possible paths, storing the path length of each, and then choosing the shortest. Of course, if the picture changes while the robot is en route, then the route may have to be refigured; that is unavoidable.

"Path-optimization" is the process of trimming out all retracing of steps from the path. A method suggested for this process is to write a list which is maintained throughout a given pattern.

Each time a change of direction is made, *excluding* bumping into obstacles, the coordinate of that point is placed at the next available list entry. Then, each time a new coordinate is walked into, the list is checked. If the coordinate already exists in the list, *all entries following that one* are deleted. Table 1, which accompanies Figure 11, illustrates the growing and shrinking of this list, resulting finally in a clean path from A to B. The robot then follows the list, using each successive coordinate as a preliminary destination.

Table 1. Itinerary List for Path Optimization

Path number	Coordinates in list (see Figure 11)
I	0,0
II	0,0 10,0
III	0,0 10,0 10,3
III reverse	0,0 10,0 10,3
IV	0,0 10,0 10,3
V	0,0 10,0 10,3 3,3
V reverse	0,0 10,0 10,3 3,3
IV reverse	0,0 10,0 10,3 3,3
III reverse	0,0 10,0 10,3
VI	0,0 10,0 10,3 14,3
VII	0,0 10,0 10,3 14,3 14,2
VIII	0,0 10,0 10,3 14,3 14,2 17,2
IX	0,0 10,0 10,3 14,3 14,2 17,2 17,6
X	0,0 10,0 10,3 14,3 14,2 17,2 17,6 10,6

You may wonder why I do not optimize the path even further by joining points P1 and P2 with a shorter path than shown. It can be done, but we are rapidly approaching a level of complexity in our software which would be hard to implement in such a small system. Notice the importance of testing every possible path before leaving A. If we had let Robot take off up the other direction first, he would have had to travel all the way around the "world" to get to B.

All the methods we have been looking at require fairly good accuracy in calculating position and direction. The robot should have some way of "calibrating" his world map's coordinate system. For this I suggest a re-charging "base" at a fixed location whose coordinates are known to him and which provide a sufficient area for the rest of the array. This base should have some type of orienting rods or tracks to accommodate a certain amount of error produced by carpets, slippery floors, timing, etc. These would force the robot into the location necessary for re-charging the batteries. In addition to the coordinates, the robot would need to know the direction to be facing in order to make contact. This also gives him a bearing calibration.

Of course, you will want to include some means of communicating with your robot. The ability to say to the computer "You are now at coordinate 15,100" is a very desirable feature. Whether that is done via terminal, speech recognition, or dip switch is up to you. The robot should map out his world himself, but you must define the limits of that universe, since you alone know initially how far he will be able to go.

Certainly the area of navigation can be explored in much more detail than has been done here. Many of the concepts barely scratch the surface of a facet of artificial intelligence that has applications everywhere in the real world. It will be up to the pioneers of robots, to a certain extent, to develop the technology that may carry us into a new generation of transportation modes.□

NO FRILLS! NO GIMMICKS! JUST GREAT

DISCOUNTS

MAIL ORDER ONLY

HAZELTINE	
1400	$679.00
1500	995.00
Mod 1	1495.00
CENTRONICS	
779-1	954.00
779-2	995.00
700-2	1350.00
761 KSR tractor	1595.00
703 tractor	2195.00
Micro Printer	395.00
NORTHSTAR	
Horizon I assembled	1629.00
kit	1339.00
Horizon II assembled	1999.00
kit	1599.00
Disk System	589.00
TELETYPE	
Mod 43	995.00
DIGITAL SYSTEMS	
Computer	$4345.00
Double Density Dual Drive	2433.00
IMSAI	
VDP 80/1000	$5895.00
VDP 44	4195.00
16K Memory assem.	399.00
PCS 80/15	679.00
15% off on all other Imsai products	
DEC	
LA 34	1149.00
CROMEMCO	
System III $1000 off	4990.00
10% off on all other Cromemco products	
TEXAS INSTRUMENTS	
810 Printer	1595.00

Most items in stock for immediate delivery. Factory-fresh, sealed cartons.

DATA DISCOUNT CENTER P.O. Box 100

135-53 Northern Blvd., Flushing, New York 11354, 212/465-6609

N.Y.S. residents add appropriate Sales Tax. Shipping FOB N.Y. BankAmericard, Master Charge add 3%. COD orders require 25% deposit.

CIRCLE INQUIRY NO. 88

MICRO-MARKET ADS SELL YOUR PRODUCTS

A new format has been established for the Micro-Market section. All ads are now 2 inches wide by 3 inches deep. Price is $200. Submit ads with check or money order to:

INTERFACE AGE Magazine
Micro-Market Ads
P.O. Box 1234
Cerritos, CA 90701

Connect your TRS-80, Apple or ANY other computer to the phone lines.

USR-330 Originate— Auto-Answer Modem



$339

- 0-300 Baud
- Stand Alone
- RS232
- 1 Year Warranty
- Crystal Controlled
- Bell 103/113
- State of the Art LSI circuitry
- 5 stage active filters

FCC certified for direct connection to phone lines via standard extension phone jack

Call or write for free literature

U.S. ROBOTICS, INC.
1035 W. LAKE ST.
CHICAGO, ILL. 60607
(312) 733-0497



SAVE **TRS-80**
10%, 15% and more on Computers.
PAN AMERICAN ELECTRONICS, INC., A
Radio Shack®
Authorized Sales Center
1117 CONWAY MISSION, TX. 78572
TOLL FREE 800/531-7466
Texas & Main No.
512/581-2765

Power Supplies **5 volts @ 25 amps**

INPUT: 115 vac 43 - 63 hz
OUTPUT: 5 vdc 0.25 amps
REGULATION: 0.1% ± 5 mV
NL-FL ± 0.1% ± 5mV for 10% Input change
RIPPLE: 2mV RMS max., 20 mV P-P max.
STABILITY: Typically 10 mV for eight hour period

SURPLUS UNITS IN LIKE NEW CONDITION



REMOTE SENSING, REMOTE VOLTAGE ADJUSTMENT, OVERLOAD PROTECTION and OVERVOLTAGE PROTECTION
ALSO AVAILABLE IN 12 VDC @ 13 AMPS $79.00 per unit

79.⁰⁰ per unit

TANTALUM CAPS
• 1 MFD 35 VDC 20%
5 for $1.00
Large Quantities Available

22/44 EDGEBOARD CONNECTOR
TIN SOLDERTAIL .156" x .200"
LARGE QUANTITIES AVAILABLE
$1.35 each 10 for $12.50

1½ AMP FULL WAVE BRIDGE RECTIFIER
400 PIV
2 for $1.00

TO-3 HOUSE MARKED DARLINGTON TRANSISTOR
$1.50 each
16 Amps 60 Volts 150 Watts
Similar to: MJ4030 PNP
Similar to: MJ4033 NPN
Specify NPN or PNP

C-H D.P.D.T. PRINTED CIRCUIT MINI TOGGLE
$1.50 each
10 for $12.50
• 6 amps @ 125 vac
• on - on

ALL ELECTRONICS CORP.
905 S. Vermont Ave. • Los Angeles, Calif. 90006
DEPT. IF-1 (213) 380-8000

STORE & WAREHOUSE HOURS
Mon. - Fri. 9 AM - 5 PM
Saturday 10 AM - 3 PM

TERMS
• Quantities Limited
• Min. Order $10.00
• Add $1.50 Shipping USA
• Calif. Res. Add 6%
• Prompt Shipping

SEND FOR OUR FREE CATALOG

Translators with computer savvy needed to translate programs and documentation from English into other languages (human, not computer). If you know micros and are fluent in French, German, Spanish, or some other tongue, we'll pay for your translation of our software. Send details and proof of language skills (a vita will help) to:

Editorial Director
Hayden Book Company, Inc.
50 Essex Street
Rochelle Park, NJ 07662



NEW Z-80 SINGLE BOARD COMPUTER

ELIMINATE Card Cases, Multiple Boards, and Bulky Power Supplies. LSI programmable SINGLE BOARD SYSTEMS with 64K, SERIAL, PARALLEL, PRINTER, VIDEO and DISK CONTROLLERS, CLOCK, CALENDAR, TIMERS, DMA, etc. 3000 PROGRAMS. Purchase SBC or systems.

(314) 961-2229

DATA WORLD, INC.
7541 Ravensridge Dr.
St. Louis, Mo. 63119



A2D/Apple II Joy Stick for computer games and graphics. Plug-in compatibility for Apple II I/O connectors. Precision open gimbaled self centering action in heavy gold vinyl clad aluminum case (not vac formed). Nice hand-held design with two game control push buttons. Our open gimbal has less drag than bale sticks – was really developed for championship acrobatic R/C flying. Comes with 6' long cable. Retail $45.00. Send check, Money Order or $5.00 cash and balance C.O.D. Master Charge and Visa OK. Postpaid. About 500 have been in use for one year. Dealer inquiries invited, both foreign and domestic.

A2D COMPANY
PO Box 6471, Greenville, S.C. 29606
Ph. 803-297-0552

SAVE MORE THAN 20%!
NORTH STAR - INTERTUBE
THINKER TOYS - MICROTEK
The smartest computers at the smartest price
Quad & Double Density

	LIST	ONLY
HORIZON-1-32K-D kit	$1,999	$1585
HORIZON-2-32K-D kit	2399	1905
Assembled & tested	2765	2195
HORIZON-2-32K kit QUAD	2799	2225
Assembled & tested	3215	2555
Pascal for North Star on Disk		49
Powerful North Star BASIC		FREE
TEI PT 212 Computer 5 MHz	8000	6250
Thinker Toys DISCUS 2/D, A&T	1149	949
Discus/2+2 1.2 megabytes, A&T	1549	1299
Measurement System Memory, A&T, 4MHz 64K		640
Godbout Memory	Call for Price	
INTERTUBE II Smart Terminal	995	780
MICROTEK Printer	750	675
ANADEX Printer	995	875
Florida Data Printer, 600 cps, 4300	Call for Price	
MARYELLEN Word Processor	Your Best Buy	38
Textwriter III		125
EZ-80 Tutorial, Learn Machine Language		25
PDS for North Star, Better than CP/M		99
Compiler for Horizon Secret Superfast Code		100
10• off software prices with computers		
VERBATIM the best Diskettes	Box of 10	29
Which Computers are best? BROCHURE		FREE
North Star documentation refundable w/HRZ		20

AMERICAN SQUARE COMPUTERS
Kivett Dr Jamestown, NC 27282 919-883-1105

GET Paid for using your Computer

FUN! Easy

RUSH COUPON FOR FREE FACTS

GREAT SPARE TIME

Send today to — DAR-IA4
3110 Fulton Ave, Sacramento CA 95821

CASH

NAME

STREET

CITY

STATE ZIP

MICRODEX

MICRO INDEX CLASSIFIED

TRS-80 HANGMAN cassette for 16K Level II. Guess the words before the hangman is drawn and you win! Send $10 to Brendan Conner, 11132 S. Hoyne, Chicago, IL 60643.

COMPUTER MAGAZINE ARTICLE INDEX — Over 1,000 categories classifies all mini-micro magazines 1973 to present. $24 includes 1980 quarterly update. MAGDEX, Box 13156I, Savannah, GA 31401.

TRS-80 LEVEL II, 16K owners: increase your profits with this inexpensive rapid-entry time study program. Free information: OMNI SYSTEMS CO., Dept. I1, Box 29347, Minneapolis, MN 55430.

IMSAI VDP-80, 64K, one megabyte disk storage IMDOS, CBASIC, AJ841 Selectric terminal, Bytesaver, asking $4995. Atlanta area, 3851 Tanglewood Rd., Snellville, GA 30278, (404) 972-9933.

SOLAR ENERGY programs for TI-58/59 calculate direct, indirect and reflected insolation for any time, date and location. $25 for program listings and descriptions. Send 3 magnetic cards with order for recoding. F STITELER, 2274 Minneola Rd., Clearwater, FL 33516.

NORTH STAR professional business software; General Ledger; Accounts Receivable; Job Bidding/Costing; Apartment Management; Inventory. $125 each. Write LADCO DEVELOPMENT CO., 1020 NE Town Terrace, Jensen Beach, FL 33457.

TRS-80 SOFTWARE. Quizzes $5 each. Quizzes include math, English, sports, etc. Games for TRS-80 $40 for a set of 10. Games by mail $13 for first 5 turns and rules. WIZARDS ARSENAL, Box 20183, Lansing, MI 48901.

ADDS 580 terminal/keyboard 24 line/80 character, 12" screen upper case, EIA and current loop 110-9600 baud. Used. Working. Manual. $450. Dan Henderson, (303) 545-5173.

APPLICATION SUBSYSTEM. Comprehensive, tested, documented, perform data base creation, maintenance and create multi reports about your family's genealogy. N*BASIC. Specify minidisk or hardcopy, only $45. BIOCHARTS, Box 423, Nanuet, NY 10954.

TRS-80 QUALITY SOFTWARE. Data base manager/report generator: no user programming, $69, MOD-II $199. A/R, $69, MOD-II $149. Inventory, key random access, $99, MOD-II $149. Word processor, $49. Mailing list, $59, MOD-II $99. Manual $5, MOD-II $10. Micro Architect, 96 Dothan St., Arlington, MA 02174.

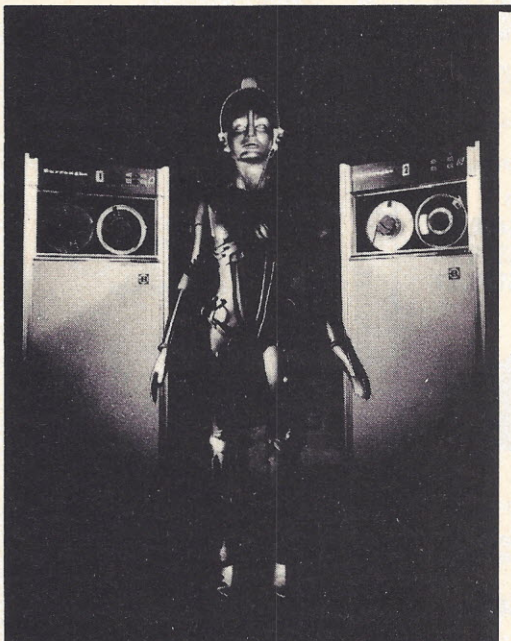
MULTIPLAYER WARGAME for TRS-80 16K-L2. Production, missiles, ABM's, bombers, submarines, anti-submarine planes, spies, video combat in thoroughly playtested WWIII game. $20. YADUK SOFTWARE, 68 Mohawk Trail, Greenfield, MA 01301.

STRUCTURAL DESIGN. Steel truss analysis and design system for North Star Horizon with floppies, CP/M and Microsoft FORTRAN. Includes programs for Warren/Pratt and general configuration trusses together with over 500 rolled steel shapes, and combination of shapes, physical properties data base. Send $10 for documentation, sample designs, information on other programs. Shapler Associates, 1959 Chalice Way, Toledo, OH 43613.

Apple-VTR Controller. Integrate CAI & VTR. Complete system $390. CAVRI Systems, 26 Trumbull Street, New Haven, CT 06511, or call Dr. Marc Schwartz, (203) 562-9873.

FOR SALE: IMSAI 8080 mainframe with 22 slot motherboard, IMSAI MIO board, 32K 250ns. memory, North Star minidisk and controller. Best offer. Call evenings at (305) 920-6545, or write to Overtone, Inc., 1443 Washington Street, Hollywood, FL 33020.

MICRODEX is a classified advertising section available to readers wishing to buy, sell or trade hardware and software. Price is $1 per word, with a 25 word minimum. The first two words are printed bold. Additional bold type is $5 per word. Send copy with check or money order to INTERFACE AGE, Dept. M-I, 16704 Marquardt Ave., Cerritos, CA 90701.



COME OUT OF THE DARK

Step into the bright, exciting world of computing. For businessmen, professionals and students.

INTERFACE AGE — a computer magazine you can understand. Written and edited for those who want to get more out of their computer.

Keep up to date with the latest new products and developments. Join the 85,000 who make reading INTERFACE AGE a priority each month.

INTERFACE AGE MAGAZINE

Please enter my subscription to INTERFACE AGE for:

- ☐ 1 year U.S. $18.00
- ☐ 1 year Canada/Mexico $20.00
- ☐ 1 year International Surface Mail $28.00
- ☐ 2 years U.S. $30.00
- ☐ 2 years Canada/Mexico $34.00
- ☐ 1 year International Air Mail $50.00

Make Check or Money Order (U.S. funds drawn on U.S. bank) payable to:
INTERFACE AGE Magazine P.O. Box 1234, Cerritos, CA 90701 or
INTERFACE AGE Europe Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany

Charge my: ☐ Visa Card ☐ Master Charge ☐ American Express Exp. Date_____

Card No. _____ Signature _____

Address _____

City _____ State_____ Zip_____ Country_____

Allow 6 to 8 weeks for delivery

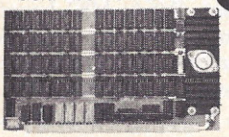
WANT TO BE KING OF THE HILL?

Treat Yourself Royally with GIMIX Unique and Incomparable Boards and Systems... DIP-switch Versatility for use with both SS50 (6800) and SS50C (6809) Systems (SWTP. etc.)

32K STATIC RAM BOARD

- SS50C Extended Addressing (can be disabled)
- 4 separate 8K blocks
- Low Power 2114L RAMS (2 AMP TYP. for 32K)
- Write Protect
- Fully Socketed for 32K
- Gold Bus Connectors

NEW!



16K . . $328.12
24K . . $438.14
32K . . $548.15

16 & 24K Versions are socketed for 32K and require only additional 2114S for expansion.

All GIMIX Memory Boards are assembled, burned-in, and tested at 2MHz.

FACTORY PRIME STATIC RAMS

2114L 450 ns . . . **$5.90** 200 ns $6.90
4044 450 ns **$5.90** 250 ns $6.90
Add $5.00 Handling on Orders Under $200.00

THE UNIQUE GIMIX 80 x 24 VIDEO BOARD

- ★ Upper and Lower Case with Descenders
- ★ Contiguous 8x10 Character Cells
- ★ Hardware Scrolling
- ★ X-Y Addressable Hardware Cursor

It is the ***ONLY*** **Video Board** that gives you **Software Control** of:

- A programmable RAM Character Generator plus 2 EPROM Character Generators (128 char. ea.).
- Selecting 256 Displayable Characters from 384 available.
- Normal or inverse video, full or reduced intensity, or combinations of these by ***both*** ASCII Code and Bit 8.
- GHOSTability — multiple boards at the same address.

Fully decoded, occupies only 2K of address space
Fully socketed — Gold bus connectors.
Assembled, Burned-in, and Tested at 2MHz.

Deluxe Version **$458.76**
Without RAM Character Generator . . **$398.24**
Other Video Boards from **$198.71**

THE CLASSY CHASSIS



- Ferro-resonant Power Supply
- Heavyweight Aluminum Cabinet with fan and provisions for two 5" disk drives.
- 6800/6809 Mother Board, fifteen 50 pin and 8 DIP-switch addressable 30 pin slots — Gold Plated Pins. Fully decoded

With Baud Rate Generator on Mother Board . . . **$828.19**

32K SYSTEM *Incomparable Features at a Comparable Price!* **$1,594.59**
Includes: Chassis, 6800 CPU, 32K RAM Board, Choice of I/O Card.

16K Version of above $1,374.49

Phone, write, or see your dealer for details and prices on our broad range of Boards and Systems for the SS50/SS50C bus and our AC Power Control Products for all computers.

GIMIX inc.

1337 W. 37th Place • Chicago, IL 60609
(312) 927-5510 • TWX 910-221-4055

The Company that delivers.
Quality Electronic products since 1975.

GIMIX® and **GHOST®** are Registered Trademarks of GIMIX Inc.

ADVERTISER INDEX

Info Inquiry Number		Page
1	A.E.I.	32
2	Altos	86-87
4	Anadex	IBC
5	Asyst Design Services	26
6	Bits Inc.	58
8	Compumax	45
9	CompuServe	93
*	The Computer Factory	32
11	Computer Furniture & Accessories, Inc.	22
12	Computer Pathways Unltd., Inc.	75
13	Compu/Time	38
14	Computronics Inc.	65
15	Cromemco Inc.	1
*	Cybernetics Inc.	24
16	Cybernetic Micro Systems	4
17	D-G Electronic Developments Co.	36
*	Data Dynamics Technology	104-105
*	DDT Best of INTERFACE AGE	42
18	Digital Video Systems	117
19,20	dilithium Press	46-47
21	The Discount Software Group	35
22	Disks, Etc.	106
23	Eaton	14-15
25	Electronic Control Technology	33
26	Electronic Specialists, Inc.	24
27	Esmark Inc.	8
28	FMG Corporation	54
29	FairCom	18
30	3G Co., Inc.	35
31	Gimix Inc.	144
32	Godbout Electronics	51
33	Graham Dorian	BC
34	Hardside	31
*	In Business Magazine	55
35	Industrial Micro Systems	11
36	Infosoft Systems Inc.	97
37	Information Unlimited Software Inc.	79
38	Integrand	34
*	INTERFACE AGE Subscriptions	28, 143
	Plus insert between pages 48 & 49	
39	Intersystems	6
40	Kybe	21
*	Lifeboat Associates	73, 91
41	Linca Industries, Inc.	40
42	Lobo Drives	25
43	Londe-Parker-Michaels, Inc.	37
44	Measurement Systems & Controls Inc.	16
45	Micah	38
46	Micro-Ap	61
47	Micro Applications Group	64
48	Micro Innovations Corp.	50
49	Micro Management Systems	84
*	Micropolis	56-57
50	Microsette Co.	37
51	Microtek, Inc.	3
52	Mountain Hardware, Inc.	17
*	NCC 80/AFIPS	77
53	Novation	19
*	NRI Schools	39
55	North Star Computers	9
56	Ohio Scientific	7
57	Organic Software	33
58	Osborne/McGraw-Hill	41
59	Personal Software	5
60	RLM Associates	10
61	Racet Computes	63
62	Radio Shack	29
63	Robotics Age	27
64	SWTPC	IFC
*	Shugart	12-13
65	Sigma International Inc.	26
67	Software Development & Training, Inc.	40
68	The Software Store	49
69	The Software Works	36
70,71	Structured Systems Group Inc.	80-81
72	Supersoft Associates	23
73	The Swingline Co.	6
74	Sybex	89
75	Taranto & Associates, Inc.	43
76	Tarbell Electronics	30
77	Tecmar, Inc.	34
78	Terrapin Inc.	60
*	Trenton Computer Festival	53
79	Ultra-Violet Products, Inc.	20
80	Vandata	30
81	John Wiley & Sons, Inc.	64
82	XComp, Inc.	17

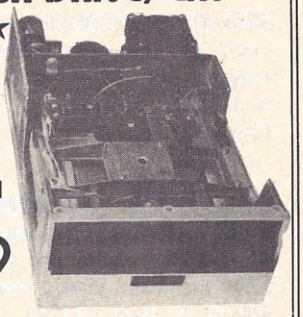
COMPUTER STORES/SURPLUS STORES

*	A2D Company	142
*	All Electronics Corp.	142
*	American Square Computers	142
83	Apparat, Inc.	133
*	Arkay Engravers, Inc.	144
84	Beta Computer Devices	123
85	Bits N Bytes, Fullerton, CA	124
86	CMC Marketing Corp.	137
87	Computer Systems Design, Inc.	106
*	DAR Sales	142
88	Data Discount Center	141
*	Data World	142
89	Futra Company	84
*	Hayden Book Co.	142
*	MTI	144
91	Microcomputer Technology Inc.	133
*	Microcomputer Technology Inc.	144
92	MicroMail	136
93	Micro Mike's, Inc.	134
94,95	Mini Micro Mart, Inc.	107, 126
96,97	Netronics R&D Ltd.	127, 129
98	Radio Hut	135
*	Radio Shack Authorized Sales Center	142
99	Sunshine Computer Inc.	131
100	T & W Communications	124
*	U.S. Robotics, Inc.	142
*	University Microfilms	144

*Manufacturer requests factory-direct inquiry.

DISK DRIVE/CRT

★SALE★



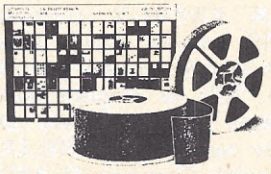
Shugart SA801R for RS MOD. II ONLY
$489

Hazeltine 1000 (unused)	$ 599
Shugart SA400	282
Pertec FD200 $282 — FD250 . . .	399
MPI B51 $279 — B52	399
SA801R w/PS/Cab	799
Dual Drives w/PS/Cab	1645

LIMITED QUANITITES

MTI 3304 W. MacArthur Blvd. Santa Ana, CA 92704
(714) 979-9923

this publication is available in microform



Please send me additional information.
University Microfilms International

300 North Zeeb Road
Dept. P.R.
Ann Arbor, MI 48106
U.S.A.

18 Bedford Row
Dept. P.R.
London, WC1R 4EJ
England

Name _____
Institution _____
Street _____
City _____
State _____ Zip _____

KEYCAPS ENGRAVED

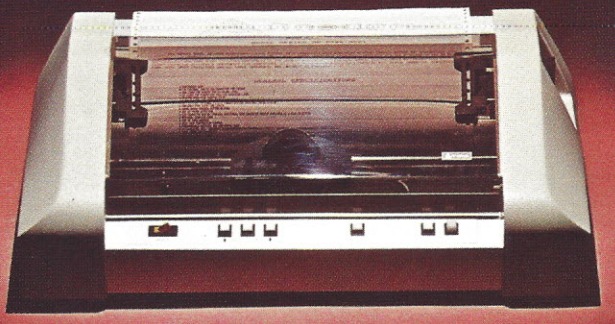
for
Cherry and Keytronic keyboards

24 Hour Service

ARKAY ENGRAVERS., INC.
2073 Newbridge Road
Bellmore, New York 11710
[516] 781-9859

Meet two new Printers from Anadex:

Resolutionary!



Introducing two totally new alphanumeric line printers from Anadex – Models DP-9500 and DP-9501 – featuring 132/175 or 132/220 columns, respectively.

Both models employ a new, Anadex-manufactured 9-wire print head with 150 million character life (optionally, 650 million) that makes them ideal for high-resolution printing requirements including high-density graphics where print quality and reliability must go hand in hand.

The full standard 96 character ASCII character set, including descenders and underlining of all upper and lower case letters, can be printed bi-directionally on up to 5 crisp copies at speeds up to 200 CPS. Adjustable-width tractors, accommodating paper from 1.75 to 15.6 inches wide, allow the printers to adapt to your application.

The three ASCII compatible interfaces (Parallel, RS-232-C, and Current Loop) are standard in every printer; so interfacing is usually a matter of "plug it in and print." With simplified interfacing, the printers also feature sophisticated communications capability including control of Vertical Spacing (6 or 8 lines/inch), Form Length and Width, Skip-Over Perforation, Auto Line Feed, and full point-to-point communications capability.

Other standard features are a 500 character FIFO buffer (optional, an additional 2048 character buffer), shortest distance sensing logic, self test, and replaceable ribbon cartridge with 6 million character life.

For complete details, attractive OEM pricing, and a demonstration, contact Anadex today.

Anadex®

ANADEX, INC. • 9825 DeSoto Avenue • Chatsworth, California 91311, U.S.A. • Telephone: (213) 998-8010 • TWX 910-494-2761

ANADEX, LTD. • Dorna House, Guildford Road • West End, Woking, Surrey GU24 9PW, England • Tel: Chobham (09905) 6333 • Telex: 858762 ANADEX G

CIRCLE INQUIRY NO. 4

Make your micro MIGHTY!



Get the most out of your microcomputer with Graham-Dorian Business software.

At any given time, your hardware is only as useful as the software you insert in it.

So it pays to rely on Graham-Dorian, the software that gets your micro performing to its fullest — almost like a mini.

Graham-Dorian, the industry leader, offers highly detailed and well-documented programs. All pretested on the job. Each so comprehensive that it takes little time to learn to run a program — even for someone who's never operated a computer before.

Programs are compatible with most major computers using CP/M disk operating systems, and come in standard 8'' or on various mini-floppy disks. Each package contains the software program in INT and BAS file form plus a user's manual and hard copy source listing. Graham-Dorian stands behind dealers with technical advice when needed.

Yes, there's a world of difference in business software. Count on Graham-Dorian for more per-package capabilities and more packages. (With new ones added every few months.)

Professional Packages

- Medical
- Dental
- Surveying
- Apartment Management
- Construction Job Costing

General Accounting Packages

- Accounts Receivable
- Accounts Payable
- General Ledger
- Inventory
- Payroll
- Cash Register

CBASIC-2

Ask your dealer for a demonstration soon.

There's A World Of Difference G/DE

Graham-Dorian Software Systems, Inc.
211 North Broadway / Wichita, KS 67202 / (316) 265-8633

CIRCLE INQUIRY NO. 33